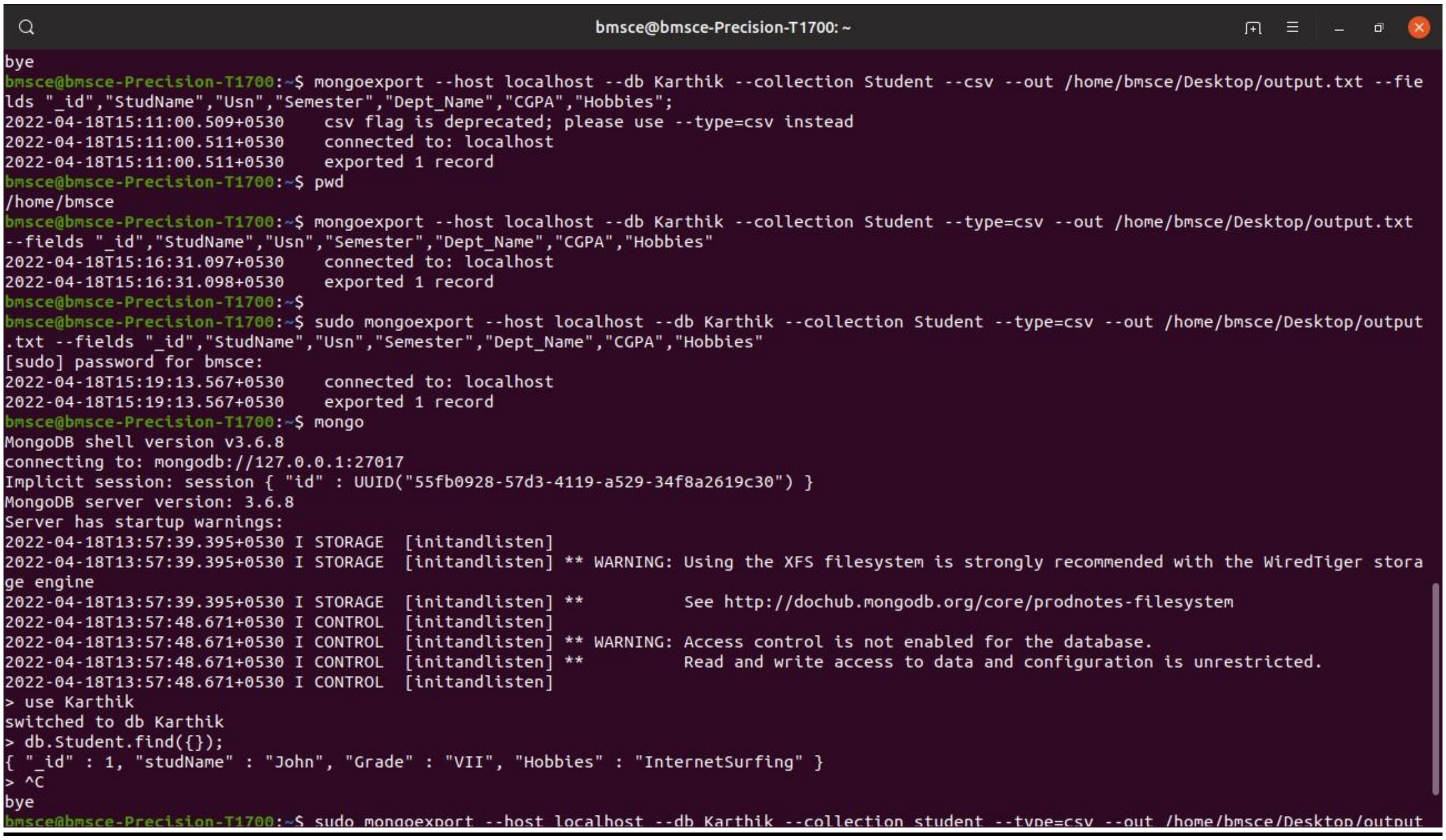
BIG DATA ANALYTICS

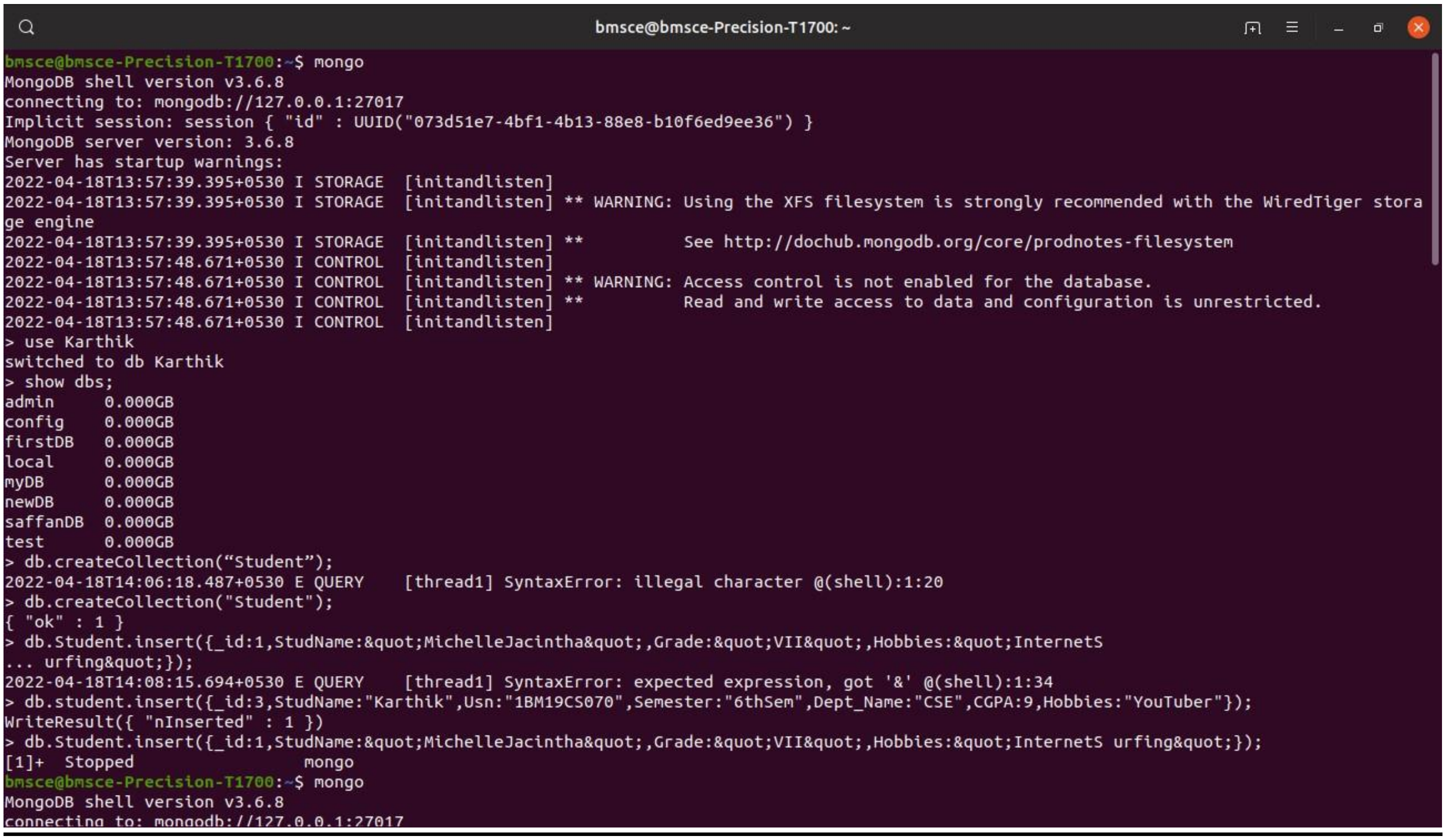
WEEK2 ASSIGNMENT

NAME – LOCHAN M R

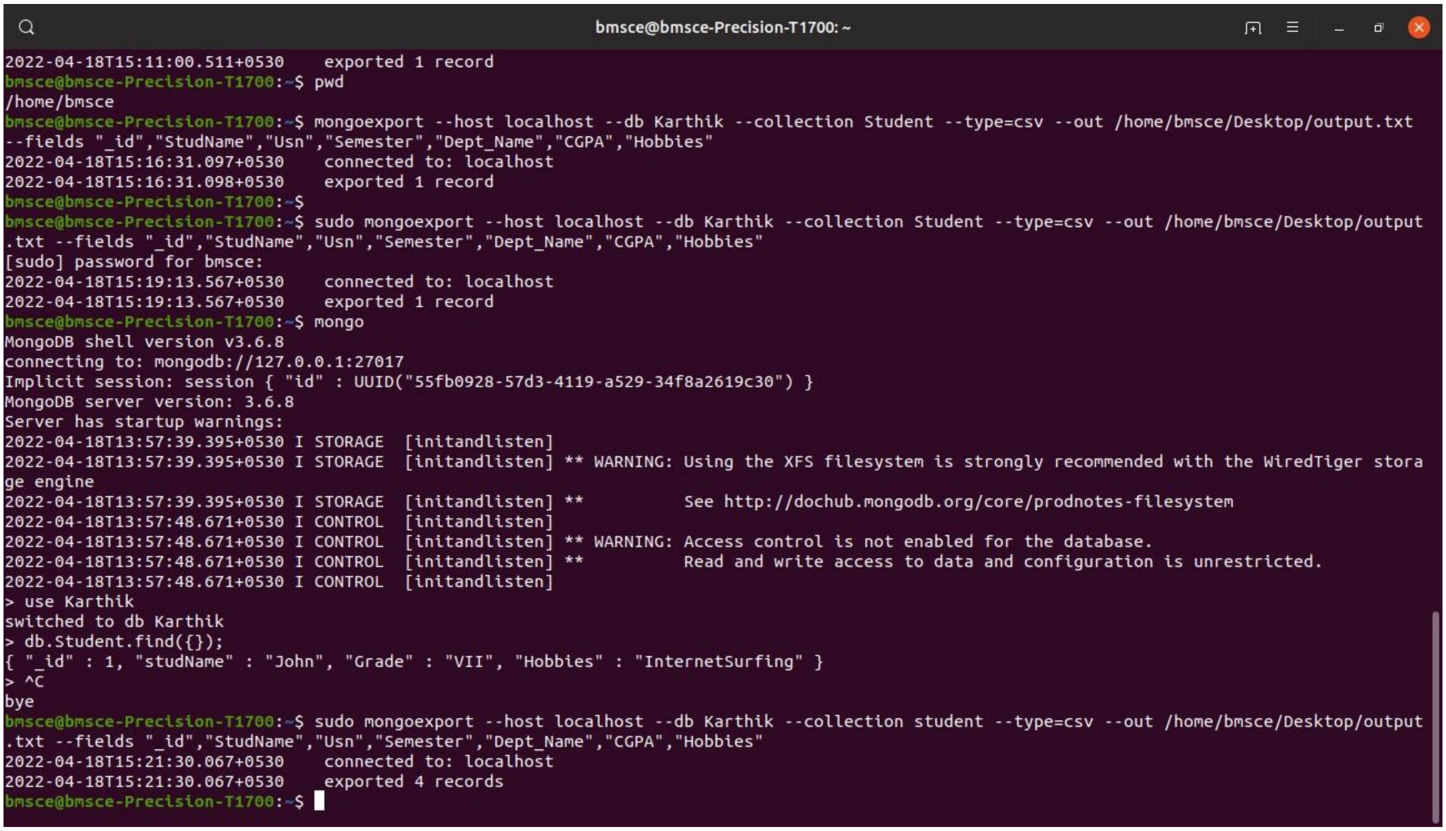
USN – 1BM20CS407



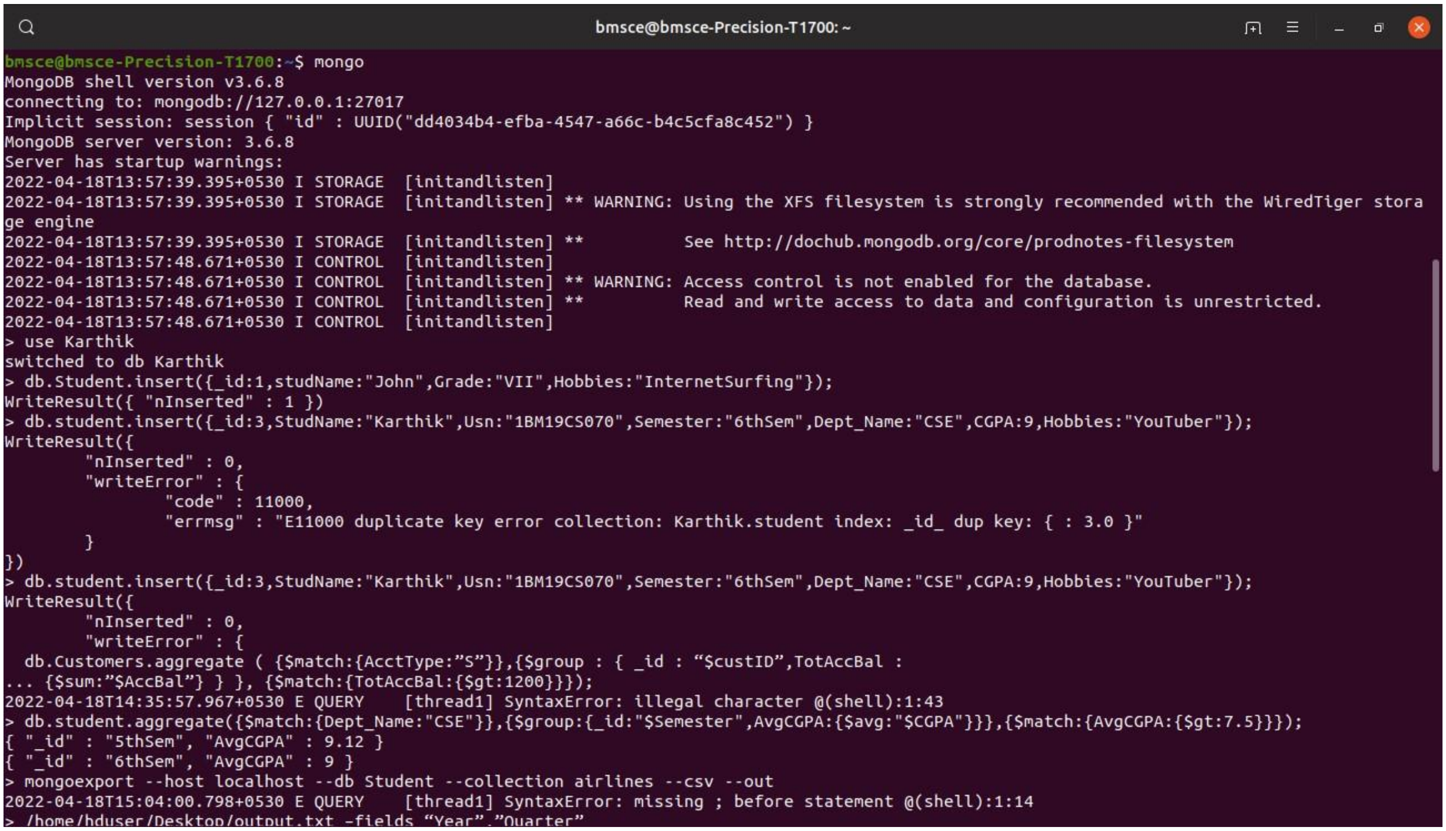
```
bye
bmsce@bmsce-Precision-T1700:~$ mongoexport --host localhost --db Karthik --collection Student --csv --out /home/bmsce/Desktop/output.txt --fie
lds "_id","StudName","Usn","Semester","Dept_Name","CGPA","Hobbies";
2022-04-18T15:11:00.509+0530    csv flag is deprecated; please use --type=csv instead
2022-04-18T15:11:00.511+0530    connected to: localhost
2022-04-18T15:11:00.511+0530    exported 1 record
bmsce@bmsce-Precision-T1700:~$ pwd
/home/bmsce
bmsce@bmsce-Precision-T1700:~$ mongoexport --host localhost --db Karthik --collection Student --type=csv --out /home/bmsce/Desktop/output.txt
--fields "_id","StudName","Usn","Semester","Dept_Name","CGPA","Hobbies"
2022-04-18T15:16:31.097+0530    connected to: localhost
2022-04-18T15:16:31.098+0530    exported 1 record
bmsce@bmsce-Precision-T1700:~$
bmsce@bmsce-Precision-T1700:~$ sudo mongoexport --host localhost --db Karthik --collection Student --type=csv --out /home/bmsce/Desktop/output
.txt --fields "_id","StudName","Usn","Semester","Dept_Name","CGPA","Hobbies"
[sudo] password for bmsce:
2022-04-18T15:19:13.567+0530    connected to: localhost
2022-04-18T15:19:13.567+0530    exported 1 record
bmsce@bmsce-Precision-T1700:~$ mongo
MongoDB shell version v3.6.8
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID("55fb0928-57d3-4119-a529-34f8a2619c30") }
MongoDB server version: 3.6.8
Server has startup warnings:
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten]
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger stora
ge engine
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
> use Karthik
switched to db Karthik
> db.Student.find({});
{ "_id" : 1, "studName" : "John", "Grade" : "VII", "Hobbies" : "InternetSurfing" }
> ^C
bye
bmsce@bmsce-Precision-T1700:~$ sudo mongoexport --host localhost --db Karthik --collection student --type=csv --out /home/bmsce/Desktop/output
```



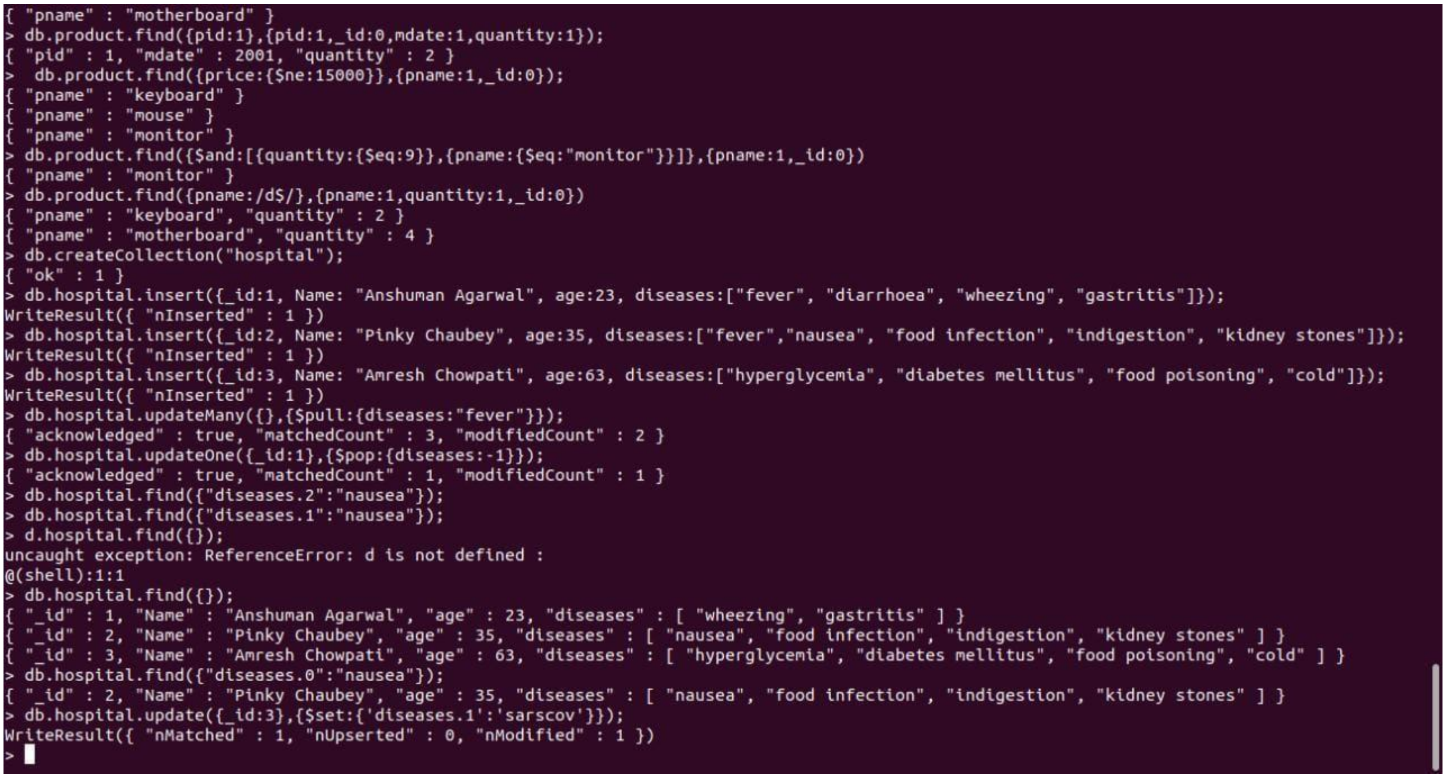
```
bmsce@bmsce-Precision-T1700:~$ mongo
MongoDB shell version v3.6.8
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID("073d51e7-4bf1-4b13-88e8-b10f6ed9ee36") }
MongoDB server version: 3.6.8
Server has startup warnings:
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten]
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger stora
ge engine
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
> use Karthik
switched to db Karthik
> show dbs;
admin     0.000GB
config    0.000GB
firstDB   0.000GB
local     0.000GB
myDB      0.000GB
newDB     0.000GB
saffanDB  0.000GB
test      0.000GB
> db.createCollection(“Student”);
2022-04-18T14:06:18.487+0530 E QUERY    [thread1] SyntaxError: illegal character @(shell):1:20
> db.createCollection("Student");
{ "ok" : 1 }
> db.Student.insert({_id:1,StudName:&quot;MichelleJacintha&quot;,Grade:&quot;VII&quot;,Hobbies:&quot;InternetS
... urfing&quot;});
2022-04-18T14:08:15.694+0530 E QUERY    [thread1] SyntaxError: expected expression, got '&' @(shell):1:34
> db.student.insert({_id:3,StudName:"Karthik",Usn:"1BM19CS070",Semester:"6thSem",Dept_Name:"CSE",CGPA:9,Hobbies:"YouTuber"});
WriteResult({ "nInserted" : 1 })
> db.Student.insert({_id:1,StudName:&quot;MichelleJacintha&quot;,Grade:&quot;VII&quot;,Hobbies:&quot;InternetS urfing&quot;});
[1]+  Stopped                 mongo
bmsce@bmsce-Precision-T1700:~$ mongo
MongoDB shell version v3.6.8
connecting to: mongodb://127.0.0.1:27017
```



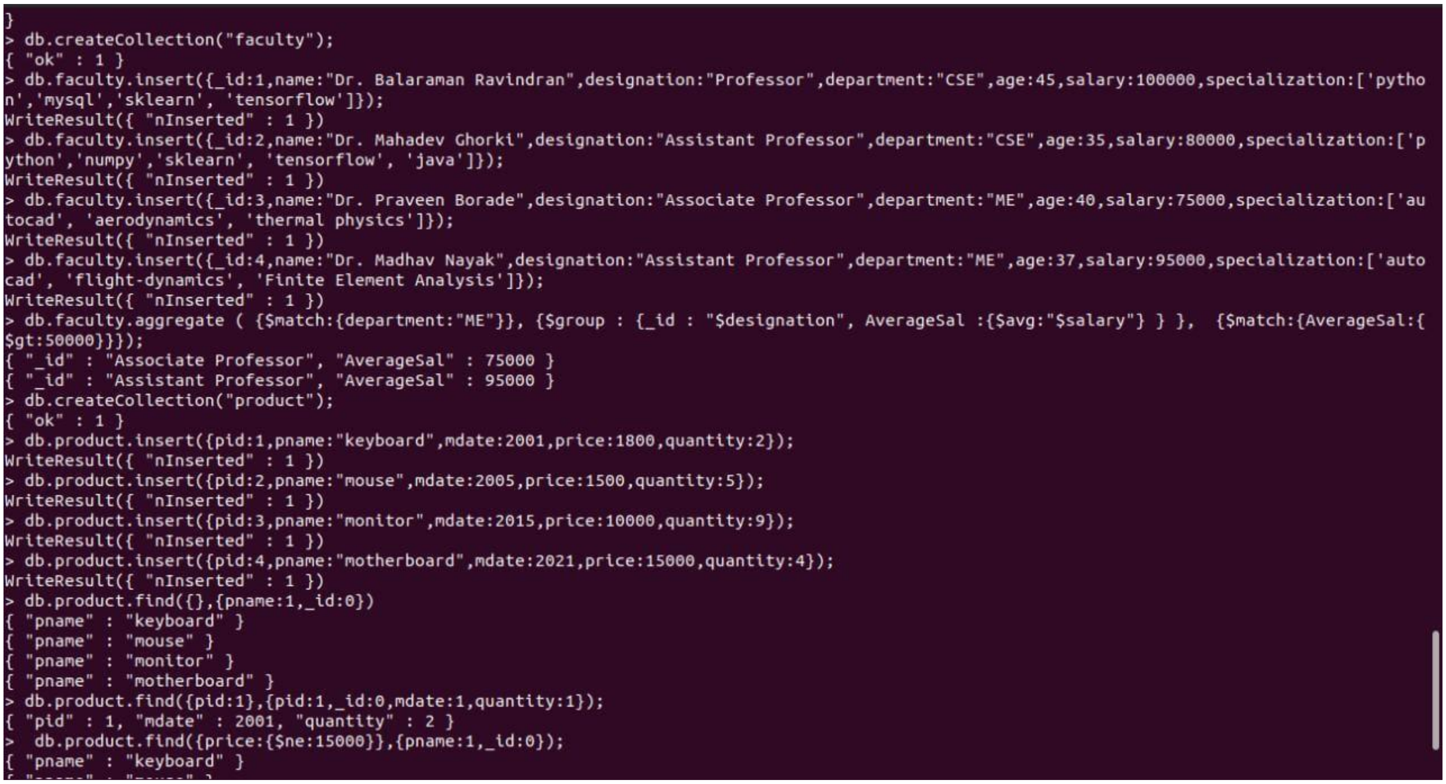
```
2022-04-18T15:11:00.511+0530    exported 1 record
bmsce@bmsce-Precision-T1700:~$ pwd
/home/bmsce
bmsce@bmsce-Precision-T1700:~$ mongoexport --host localhost --db Karthik --collection Student --type=csv --out /home/bmsce/Desktop/output.txt
--fields "_id","StudName","Usn","Semester","Dept_Name","CGPA","Hobbies"
2022-04-18T15:16:31.097+0530    connected to: localhost
2022-04-18T15:16:31.098+0530    exported 1 record
bmsce@bmsce-Precision-T1700:~$
bmsce@bmsce-Precision-T1700:~$ sudo mongoexport --host localhost --db Karthik --collection Student --type=csv --out /home/bmsce/Desktop/output
.txt --fields "_id","StudName","Usn","Semester","Dept_Name","CGPA","Hobbies"
[sudo] password for bmsce:
2022-04-18T15:19:13.567+0530    connected to: localhost
2022-04-18T15:19:13.567+0530    exported 1 record
bmsce@bmsce-Precision-T1700:~$ mongo
MongoDB shell version v3.6.8
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID("55fb0928-57d3-4119-a529-34f8a2619c30") }
MongoDB server version: 3.6.8
Server has startup warnings:
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten]
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger stora
ge engine
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
> use Karthik
switched to db Karthik
> db.Student.find({});
{ "_id" : 1, "studName" : "John", "Grade" : "VII", "Hobbies" : "InternetSurfing" }
> ^C
bye
bmsce@bmsce-Precision-T1700:~$ sudo mongoexport --host localhost --db Karthik --collection student --type=csv --out /home/bmsce/Desktop/output
.txt --fields "_id","StudName","Usn","Semester","Dept_Name","CGPA","Hobbies"
2022-04-18T15:21:30.067+0530    connected to: localhost
2022-04-18T15:21:30.067+0530    exported 4 records
bmsce@bmsce-Precision-T1700:~$
```



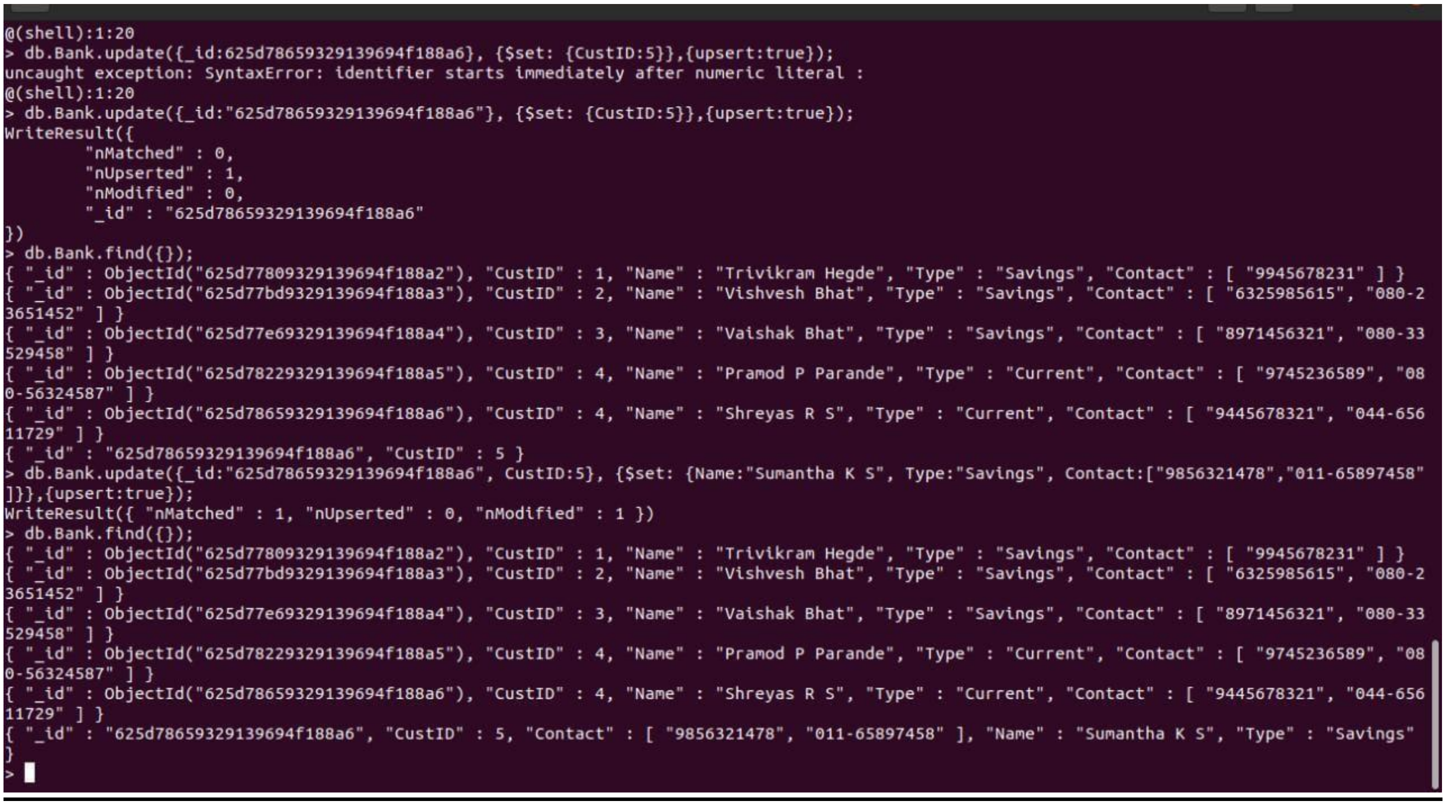
```
bmsce@bmsce-Precision-T1700:~$ mongo
MongoDB shell version v3.6.8
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID("dd4034b4-efba-4547-a66c-b4c5cfa8c452") }
MongoDB server version: 3.6.8
Server has startup warnings:
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten]
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger stora
ge engine
2022-04-18T13:57:39.395+0530 I STORAGE  [initandlisten] **          See http://dochub.mongodb.org/core/prodnotes-filesystem
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten] **          Read and write access to data and configuration is unrestricted.
2022-04-18T13:57:48.671+0530 I CONTROL  [initandlisten]
> use Karthik
switched to db Karthik
> db.Student.insert({_id:1,studName:"John",Grade:"VII",Hobbies:"InternetSurfing"});
WriteResult({ "nInserted" : 1 })
> db.student.insert({_id:3,StudName:"Karthik",Usn:"1BM19CS070",Semester:"6thSem",Dept_Name:"CSE",CGPA:9,Hobbies:"YouTuber"});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
                "code" : 11000,
                "errmsg" : "E11000 duplicate key error collection: Karthik.student index: _id_ dup key: { : 3.0 }"
        }
})
> db.student.insert({_id:3,StudName:"Karthik",Usn:"1BM19CS070",Semester:"6thSem",Dept_Name:"CSE",CGPA:9,Hobbies:"YouTuber"});
WriteResult({
        "nInserted" : 0,
        "writeError" : {
  db.Customers.aggregate ( {$match:{AcctType:”S”}},{$group : { _id : “$custID”,TotAccBal :
... {$sum:”$AccBal”} } }, {$match:{TotAccBal:{$gt:1200}}});
2022-04-18T14:35:57.967+0530 E QUERY    [thread1] SyntaxError: illegal character @(shell):1:43
> db.student.aggregate({$match:{Dept_Name:"CSE"}},{$group:{_id:"$Semester",AvgCGPA:{$avg:"$CGPA"}}},{$match:{AvgCGPA:{$gt:7.5}}});
{ "_id" : "5thSem", "AvgCGPA" : 9.12 }
{ "_id" : "6thSem", "AvgCGPA" : 9 }
> mongoexport --host localhost --db Student --collection airlines --csv --out
2022-04-18T15:04:00.798+0530 E QUERY    [thread1] SyntaxError: missing ; before statement @(shell):1:14
> /home/hduser/Desktop/output.txt –fields “Year”,”Quarter”
```



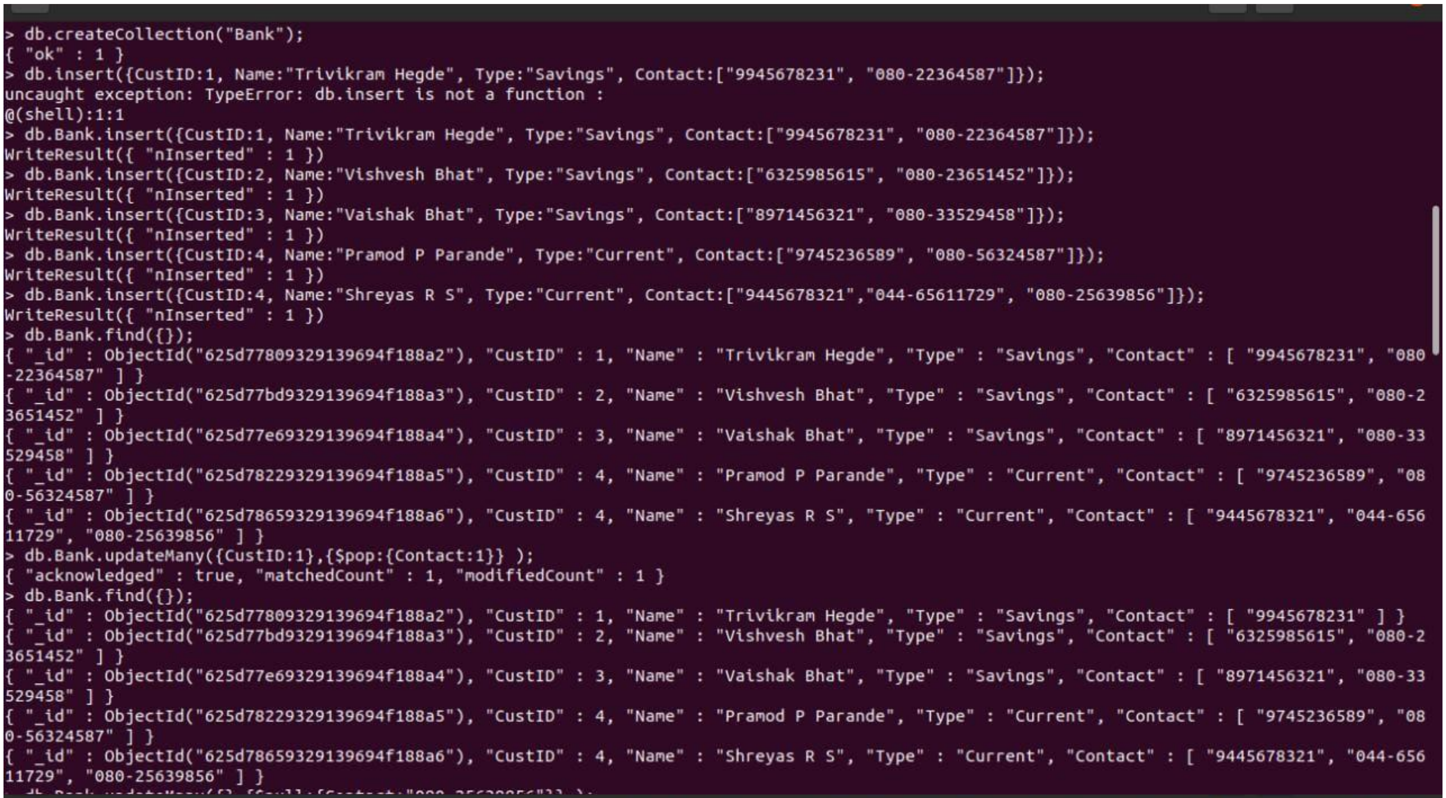
```
{ "pname" : "motherboard" }
> db.product.find({pid:1},{pid:1,_id:0,mdate:1,quantity:1});
{ "pid" : 1, "mdate" : 2001, "quantity" : 2 }
>  db.product.find({price:{$ne:15000}},{pname:1,_id:0});
{ "pname" : "keyboard" }
{ "pname" : "mouse" }
{ "pname" : "monitor" }
> db.product.find({$and:[{quantity:{$eq:9}},{pname:{$eq:"monitor"}}]},{pname:1,_id:0})
{ "pname" : "monitor" }
> db.product.find({pname:/d$/},{pname:1,quantity:1,_id:0})
{ "pname" : "keyboard", "quantity" : 2 }
{ "pname" : "motherboard", "quantity" : 4 }
> db.createCollection("hospital");
{ "ok" : 1 }
> db.hospital.insert({_id:1, Name: "Anshuman Agarwal", age:23, diseases:["fever", "diarrhoea", "wheezing", "gastritis"]});
WriteResult({ "nInserted" : 1 })
> db.hospital.insert({_id:2, Name: "Pinky Chaubey", age:35, diseases:["fever","nausea", "food infection", "indigestion", "kidney stones"]});
WriteResult({ "nInserted" : 1 })
> db.hospital.insert({_id:3, Name: "Amresh Chowpati", age:63, diseases:["hyperglycemia", "diabetes mellitus", "food poisoning", "cold"]});
WriteResult({ "nInserted" : 1 })
> db.hospital.updateMany({},{$pull:{diseases:"fever"}});
{ "acknowledged" : true, "matchedCount" : 3, "modifiedCount" : 2 }
> db.hospital.updateOne({_id:1},{$pop:{diseases:-1}});
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.hospital.find({"diseases.2":"nausea"});
> db.hospital.find({"diseases.1":"nausea"});
> d.hospital.find({});
uncaught exception: ReferenceError: d is not defined :
@(shell):1:1
> db.hospital.find({});
{ "_id" : 1, "Name" : "Anshuman Agarwal", "age" : 23, "diseases" : [ "wheezing", "gastritis" ] }
{ "_id" : 2, "Name" : "Pinky Chaubey", "age" : 35, "diseases" : [ "nausea", "food infection", "indigestion", "kidney stones" ] }
{ "_id" : 3, "Name" : "Amresh Chowpati", "age" : 63, "diseases" : [ "hyperglycemia", "diabetes mellitus", "food poisoning", "cold" ] }
> db.hospital.find({"diseases.0":"nausea"});
{ "_id" : 2, "Name" : "Pinky Chaubey", "age" : 35, "diseases" : [ "nausea", "food infection", "indigestion", "kidney stones" ] }
> db.hospital.update({_id:3},{$set:{'diseases.1':'sarscov'}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> █
```



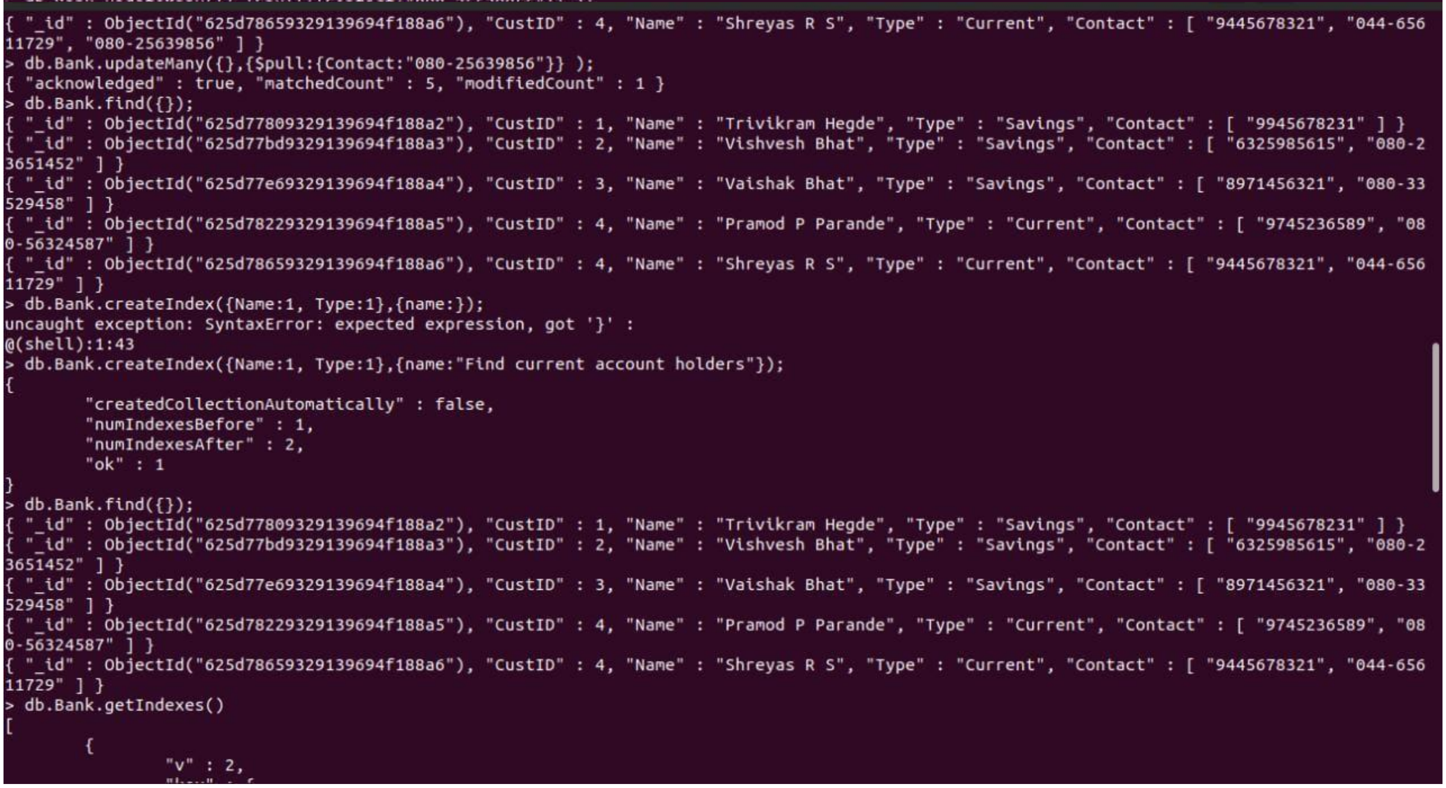
```
}
> db.createCollection("faculty");
{ "ok" : 1 }
> db.faculty.insert({_id:1,name:"Dr. Balaraman Ravindran",designation:"Professor",department:"CSE",age:45,salary:100000,specialization:['pytho
n','mysql','sklearn', 'tensorflow']});
WriteResult({ "nInserted" : 1 })
> db.faculty.insert({_id:2,name:"Dr. Mahadev Ghorki",designation:"Assistant Professor",department:"CSE",age:35,salary:80000,specialization:['p
ython','numpy','sklearn', 'tensorflow', 'java']});
WriteResult({ "nInserted" : 1 })
> db.faculty.insert({_id:3,name:"Dr. Praveen Borade",designation:"Associate Professor",department:"ME",age:40,salary:75000,specialization:['au
tocad', 'aerodynamics', 'thermal physics']});
WriteResult({ "nInserted" : 1 })
> db.faculty.insert({_id:4,name:"Dr. Madhav Nayak",designation:"Assistant Professor",department:"ME",age:37,salary:95000,specialization:['auto
cad', 'flight-dynamics', 'Finite Element Analysis']});
WriteResult({ "nInserted" : 1 })
> db.faculty.aggregate ( {$match:{department:"ME"}}, {$group : {_id : "$designation", AverageSal :{$avg:"$salary"} } },  {$match:{AverageSal:{
$gt:50000}}});
{ "_id" : "Associate Professor", "AverageSal" : 75000 }
{ "_id" : "Assistant Professor", "AverageSal" : 95000 }
> db.createCollection("product");
{ "ok" : 1 }
> db.product.insert({pid:1,pname:"keyboard",mdate:2001,price:1800,quantity:2});
WriteResult({ "nInserted" : 1 })
> db.product.insert({pid:2,pname:"mouse",mdate:2005,price:1500,quantity:5});
WriteResult({ "nInserted" : 1 })
> db.product.insert({pid:3,pname:"monitor",mdate:2015,price:10000,quantity:9});
WriteResult({ "nInserted" : 1 })
> db.product.insert({pid:4,pname:"motherboard",mdate:2021,price:15000,quantity:4});
WriteResult({ "nInserted" : 1 })
> db.product.find({},{pname:1,_id:0})
{ "pname" : "keyboard" }
{ "pname" : "mouse" }
{ "pname" : "monitor" }
{ "pname" : "motherboard" }
> db.product.find({pid:1},{pid:1,_id:0,mdate:1,quantity:1});
{ "pid" : 1, "mdate" : 2001, "quantity" : 2 }
>  db.product.find({price:{$ne:15000}},{pname:1,_id:0});
{ "pname" : "keyboard" }
```



```
@(shell):1:20
> db.Bank.update({_id:625d78659329139694f188a6}, {$set: {CustID:5}},{upsert:true});
uncaught exception: SyntaxError: identifier starts immediately after numeric literal :
@(shell):1:20
> db.Bank.update({_id:"625d78659329139694f188a6"}, {$set: {CustID:5}},{upsert:true});
WriteResult({
        "nMatched" : 0,
        "nUpserted" : 1,
        "nModified" : 0,
        "_id" : "625d78659329139694f188a6"
})
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
{ "_id" : "625d78659329139694f188a6", "CustID" : 5 }
> db.Bank.update({_id:"625d78659329139694f188a6", CustID:5}, {$set: {Name:"Sumantha K S", Type:"Savings", Contact:["9856321478","011-65897458"
]}},{upsert:true});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
{ "_id" : "625d78659329139694f188a6", "CustID" : 5, "Contact" : [ "9856321478", "011-65897458" ], "Name" : "Sumantha K S", "Type" : "Savings"
}
>
```



```
> db.createCollection("Bank");
{ "ok" : 1 }
> db.insert({CustID:1, Name:"Trivikram Hegde", Type:"Savings", Contact:["9945678231", "080-22364587"]});
uncaught exception: TypeError: db.insert is not a function :
@(shell):1:1
> db.Bank.insert({CustID:1, Name:"Trivikram Hegde", Type:"Savings", Contact:["9945678231", "080-22364587"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:2, Name:"Vishvesh Bhat", Type:"Savings", Contact:["6325985615", "080-23651452"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:3, Name:"Vaishak Bhat", Type:"Savings", Contact:["8971456321", "080-33529458"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:4, Name:"Pramod P Parande", Type:"Current", Contact:["9745236589", "080-56324587"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.insert({CustID:4, Name:"Shreyas R S", Type:"Current", Contact:["9445678321","044-65611729", "080-25639856"]});
WriteResult({ "nInserted" : 1 })
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231", "080
-22364587" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729", "080-25639856" ] }
> db.Bank.updateMany({CustID:1},{$pop:{Contact:1}} );
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729", "080-25639856" ] }
```



```
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729", "080-25639856" ] }
> db.Bank.updateMany({},{$pull:{Contact:"080-25639856"}} );
{ "acknowledged" : true, "matchedCount" : 5, "modifiedCount" : 1 }
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
> db.Bank.createIndex({Name:1, Type:1},{name:});
uncaught exception: SyntaxError: expected expression, got '}' :
@(shell):1:43
> db.Bank.createIndex({Name:1, Type:1},{name:"Find current account holders"});
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
> db.Bank.find({});
{ "_id" : ObjectId("625d77809329139694f188a2"), "CustID" : 1, "Name" : "Trivikram Hegde", "Type" : "Savings", "Contact" : [ "9945678231" ] }
{ "_id" : ObjectId("625d77bd9329139694f188a3"), "CustID" : 2, "Name" : "Vishvesh Bhat", "Type" : "Savings", "Contact" : [ "6325985615", "080-2
3651452" ] }
{ "_id" : ObjectId("625d77e69329139694f188a4"), "CustID" : 3, "Name" : "Vaishak Bhat", "Type" : "Savings", "Contact" : [ "8971456321", "080-33
529458" ] }
{ "_id" : ObjectId("625d78229329139694f188a5"), "CustID" : 4, "Name" : "Pramod P Parande", "Type" : "Current", "Contact" : [ "9745236589", "08
0-56324587" ] }
{ "_id" : ObjectId("625d78659329139694f188a6"), "CustID" : 4, "Name" : "Shreyas R S", "Type" : "Current", "Contact" : [ "9445678321", "044-656
11729" ] }
> db.Bank.getIndexes()
[
        {
                "v" : 2,
```