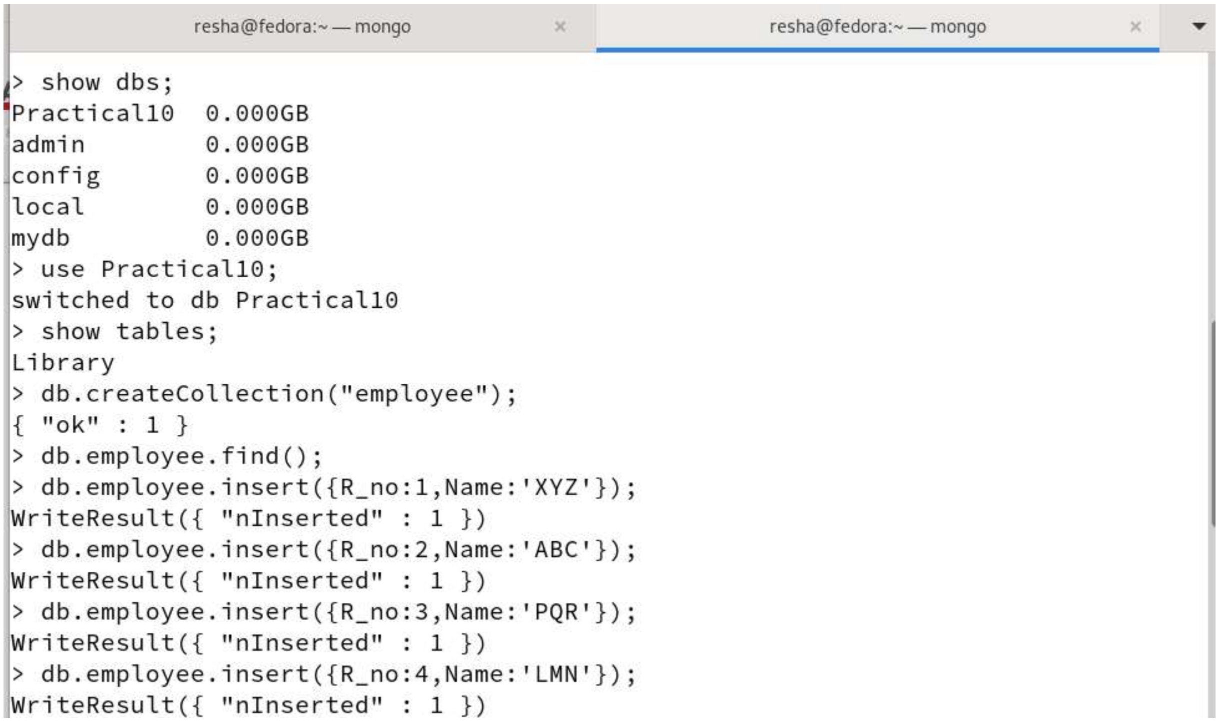
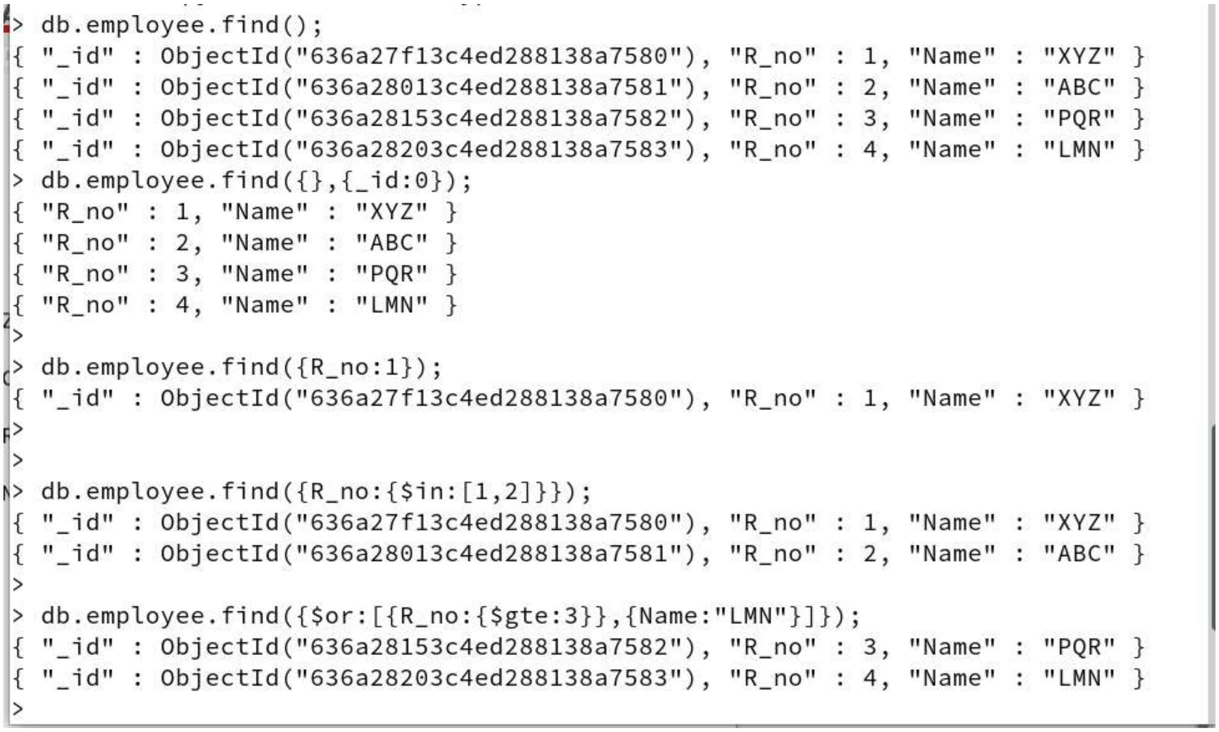
Practical No :- 10

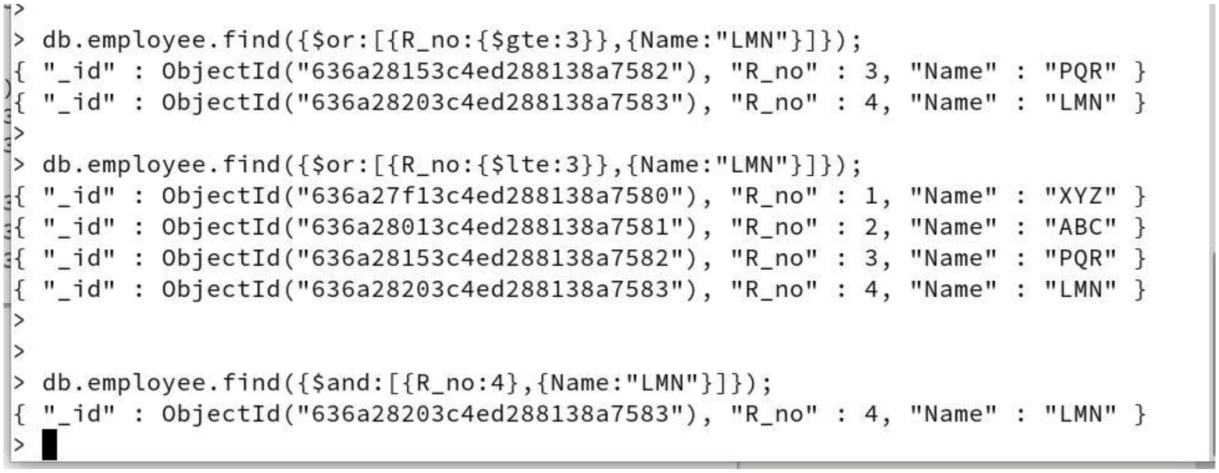
MongoDB Queries:
Design and Develop MongoDB Queries using CRUD operations.(Use CRUD operations, SAVE method, logical operators etc.).



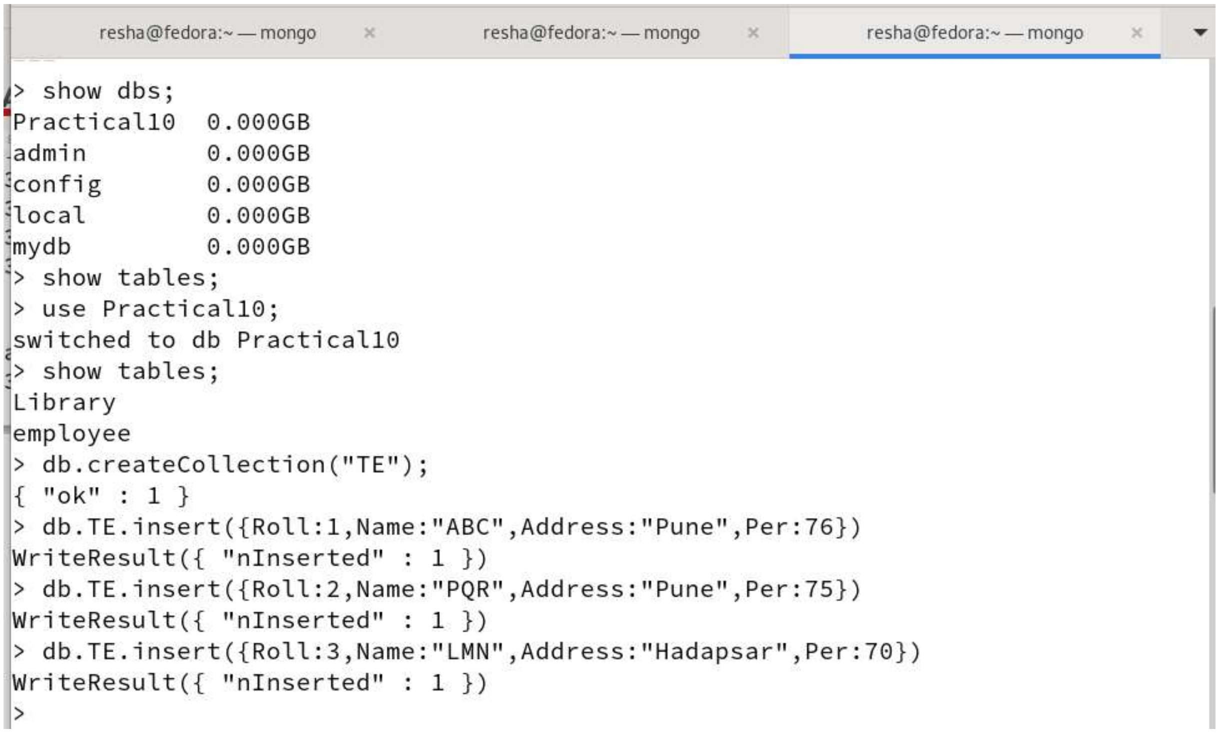
```
> show dbs;
Practical10  0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
mydb         0.000GB
> use Practical10;
switched to db Practical10
> show tables;
Library
> db.createCollection("employee");
{ "ok" : 1 }
> db.employee.find();
> db.employee.insert({R_no:1,Name:'XYZ'});
WriteResult({ "nInserted" : 1 })
> db.employee.insert({R_no:2,Name:'ABC'});
WriteResult({ "nInserted" : 1 })
> db.employee.insert({R_no:3,Name:'PQR'});
WriteResult({ "nInserted" : 1 })
> db.employee.insert({R_no:4,Name:'LMN'});
WriteResult({ "nInserted" : 1 })
```



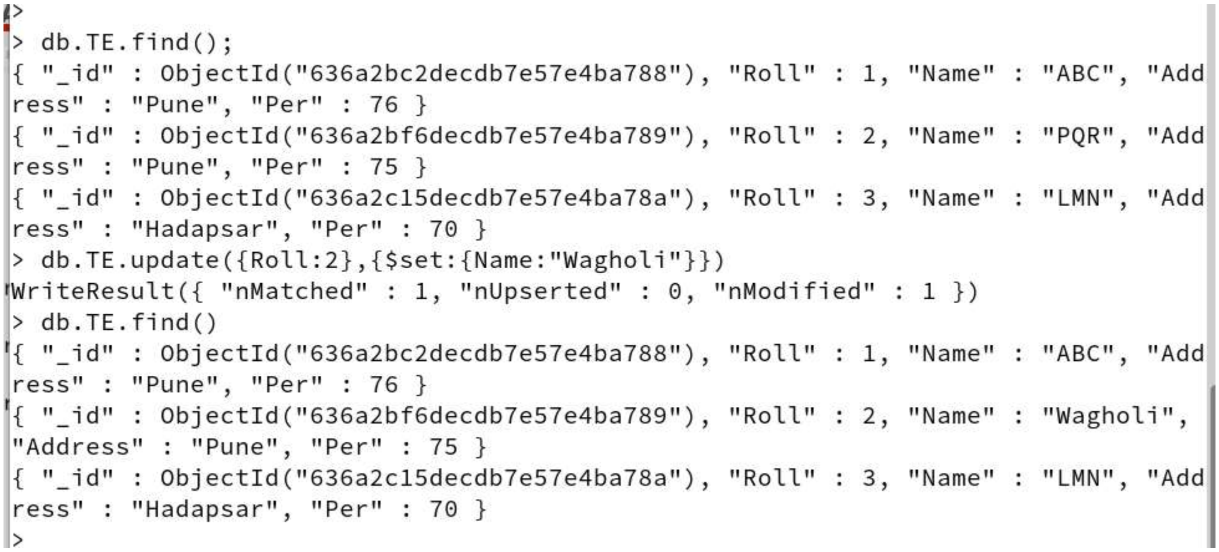
```
> db.employee.find();
{ "_id" : ObjectId("636a27f13c4ed288138a7580"), "R_no" : 1, "Name" : "XYZ" }
{ "_id" : ObjectId("636a28013c4ed288138a7581"), "R_no" : 2, "Name" : "ABC" }
{ "_id" : ObjectId("636a28153c4ed288138a7582"), "R_no" : 3, "Name" : "PQR" }
{ "_id" : ObjectId("636a28203c4ed288138a7583"), "R_no" : 4, "Name" : "LMN" }
> db.employee.find({},{_id:0});
{ "R_no" : 1, "Name" : "XYZ" }
{ "R_no" : 2, "Name" : "ABC" }
{ "R_no" : 3, "Name" : "PQR" }
{ "R_no" : 4, "Name" : "LMN" }
>
> db.employee.find({R_no:1});
{ "_id" : ObjectId("636a27f13c4ed288138a7580"), "R_no" : 1, "Name" : "XYZ" }
>
>
> db.employee.find({R_no:{$in:[1,2]}});
{ "_id" : ObjectId("636a27f13c4ed288138a7580"), "R_no" : 1, "Name" : "XYZ" }
{ "_id" : ObjectId("636a28013c4ed288138a7581"), "R_no" : 2, "Name" : "ABC" }
>
> db.employee.find({$or:[{R_no:{$gte:3}},{Name:"LMN"}]});
{ "_id" : ObjectId("636a28153c4ed288138a7582"), "R_no" : 3, "Name" : "PQR" }
{ "_id" : ObjectId("636a28203c4ed288138a7583"), "R_no" : 4, "Name" : "LMN" }
>
```



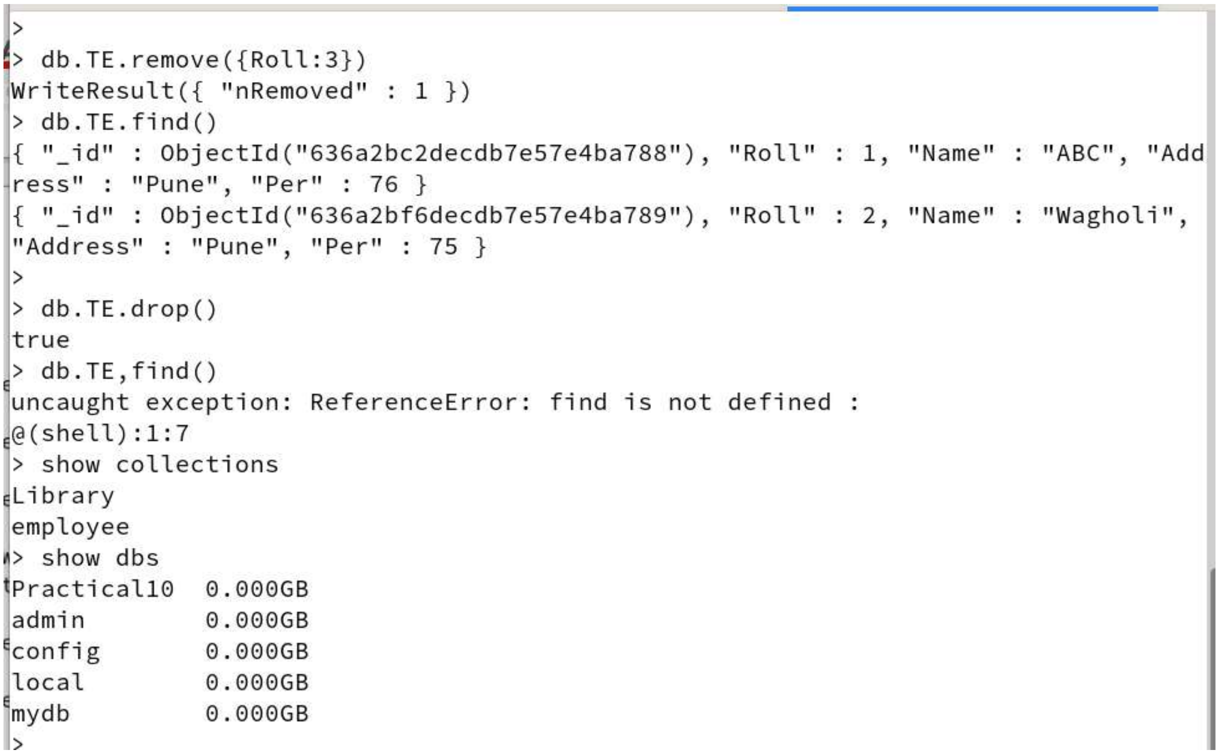
```
>
> db.employee.find({$or:[{R_no:{$gte:3}},{Name:"LMN"}]});
{ "_id" : ObjectId("636a28153c4ed288138a7582"), "R_no" : 3, "Name" : "PQR" }
{ "_id" : ObjectId("636a28203c4ed288138a7583"), "R_no" : 4, "Name" : "LMN" }
>
> db.employee.find({$or:[{R_no:{$lte:3}},{Name:"LMN"}]});
{ "_id" : ObjectId("636a27f13c4ed288138a7580"), "R_no" : 1, "Name" : "XYZ" }
{ "_id" : ObjectId("636a28013c4ed288138a7581"), "R_no" : 2, "Name" : "ABC" }
{ "_id" : ObjectId("636a28153c4ed288138a7582"), "R_no" : 3, "Name" : "PQR" }
{ "_id" : ObjectId("636a28203c4ed288138a7583"), "R_no" : 4, "Name" : "LMN" }
>
>
> db.employee.find({$and:[{R_no:4},{Name:"LMN"}]});
{ "_id" : ObjectId("636a28203c4ed288138a7583"), "R_no" : 4, "Name" : "LMN" }
> 
```



```
> show dbs;
Practical10  0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
mydb         0.000GB
> show tables;
> use Practical10;
switched to db Practical10
> show tables;
Library
employee
> db.createCollection("TE");
{ "ok" : 1 }
> db.TE.insert({Roll:1,Name:"ABC",Address:"Pune",Per:76})
WriteResult({ "nInserted" : 1 })
> db.TE.insert({Roll:2,Name:"PQR",Address:"Pune",Per:75})
WriteResult({ "nInserted" : 1 })
> db.TE.insert({Roll:3,Name:"LMN",Address:"Hadapsar",Per:70})
WriteResult({ "nInserted" : 1 })
>
```



```
>
> db.TE.find();
{ "_id" : ObjectId("636a2bc2decdb7e57e4ba788"), "Roll" : 1, "Name" : "ABC", "Add
ress" : "Pune", "Per" : 76 }
{ "_id" : ObjectId("636a2bf6decdb7e57e4ba789"), "Roll" : 2, "Name" : "PQR", "Add
ress" : "Pune", "Per" : 75 }
{ "_id" : ObjectId("636a2c15decdb7e57e4ba78a"), "Roll" : 3, "Name" : "LMN", "Add
ress" : "Hadapsar", "Per" : 70 }
> db.TE.update({Roll:2},{$set:{Name:"Wagholi"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.TE.find()
{ "_id" : ObjectId("636a2bc2decdb7e57e4ba788"), "Roll" : 1, "Name" : "ABC", "Add
ress" : "Pune", "Per" : 76 }
{ "_id" : ObjectId("636a2bf6decdb7e57e4ba789"), "Roll" : 2, "Name" : "Wagholi",
"Address" : "Pune", "Per" : 75 }
{ "_id" : ObjectId("636a2c15decdb7e57e4ba78a"), "Roll" : 3, "Name" : "LMN", "Add
ress" : "Hadapsar", "Per" : 70 }
>
```



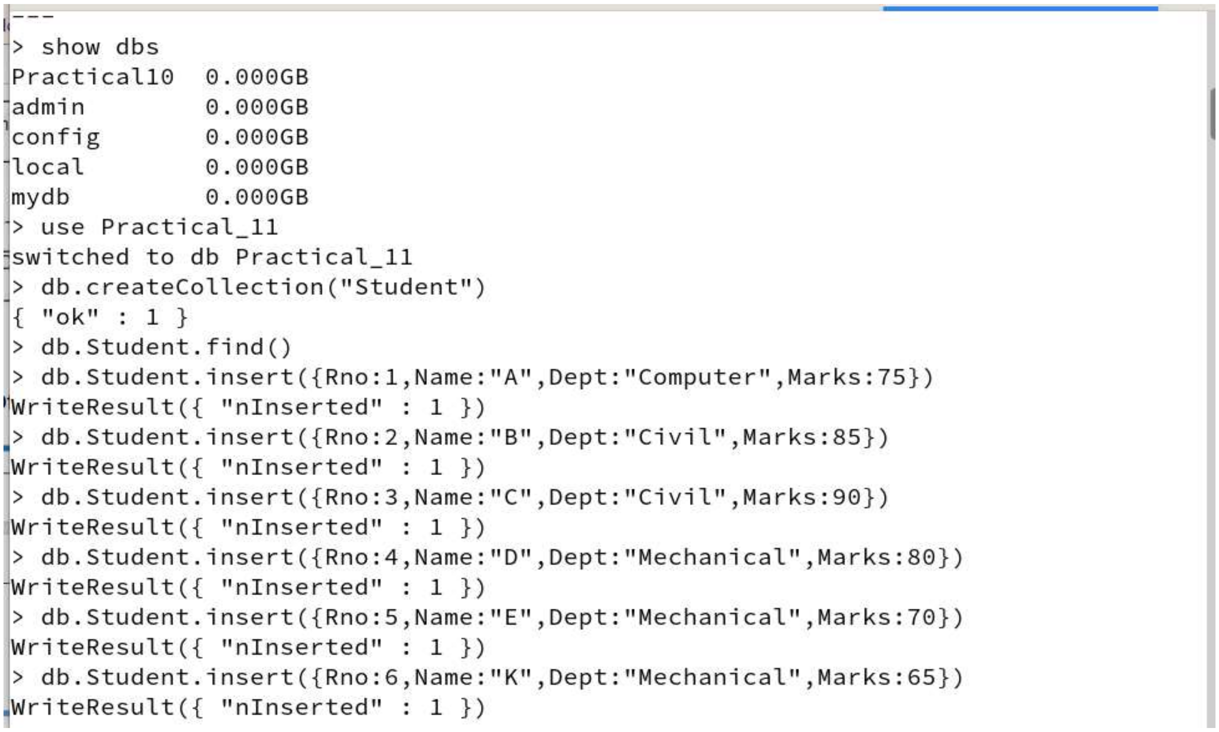
```
>
> db.TE.remove({Roll:3})
WriteResult({ "nRemoved" : 1 })
> db.TE.find()
{ "_id" : ObjectId("636a2bc2decdb7e57e4ba788"), "Roll" : 1, "Name" : "ABC", "Add
ress" : "Pune", "Per" : 76 }
{ "_id" : ObjectId("636a2bf6decdb7e57e4ba789"), "Roll" : 2, "Name" : "Wagholi",
"Address" : "Pune", "Per" : 75 }
>
> db.TE.drop()
true
> db.TE,find()
uncaught exception: ReferenceError: find is not defined :
@(shell):1:7
> show collections
Library
employee
> show dbs
Practical10  0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
mydb         0.000GB
>
```

Practical No :- 11

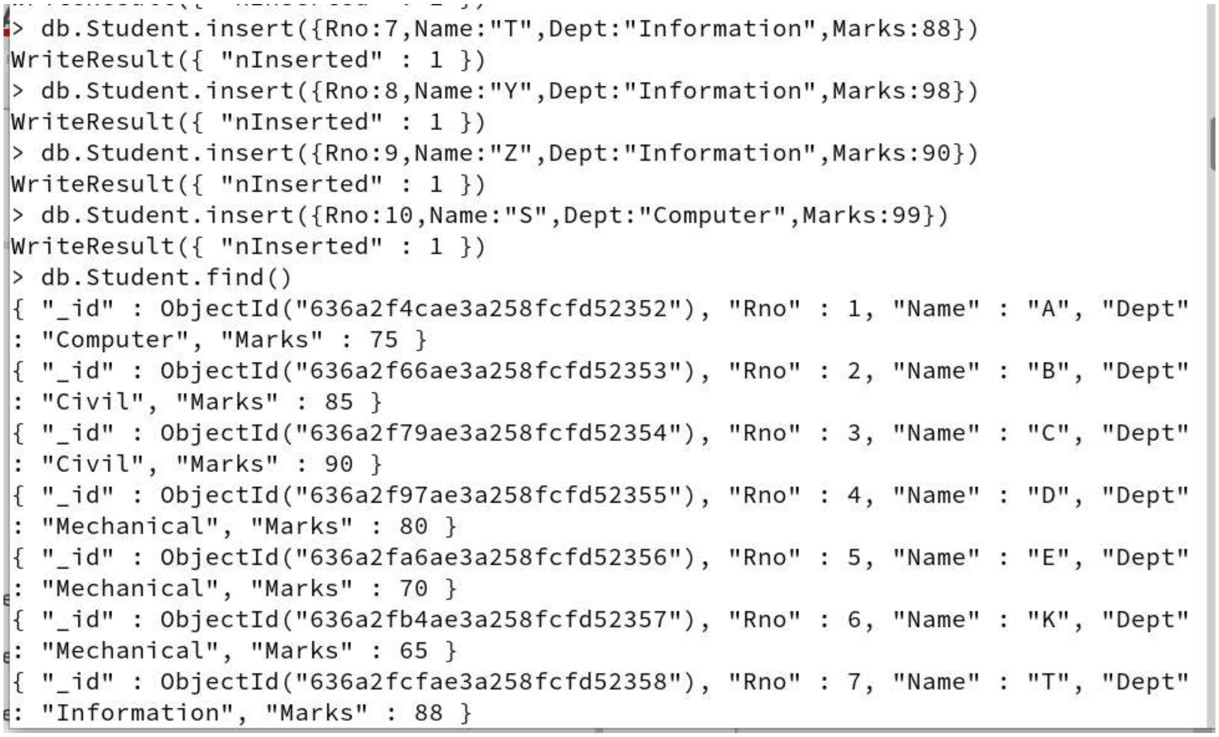
MongoDB – Aggregation and Indexing:

Design and Develop MongoDB Queries using aggregation and indexing with suitable example using MongoDB.

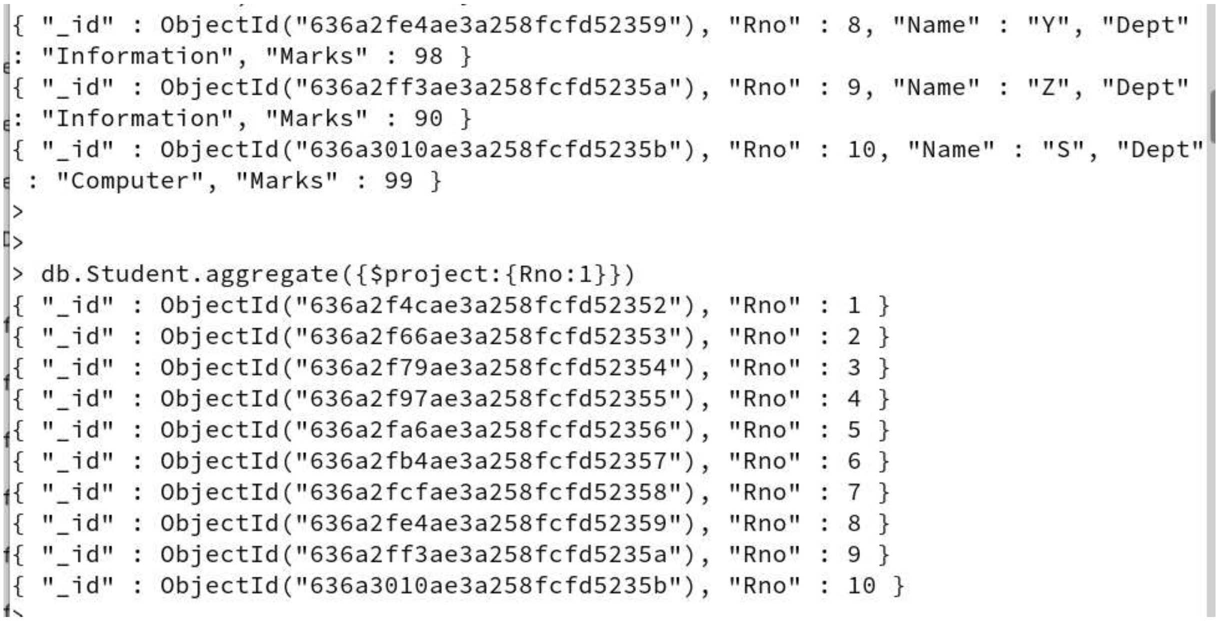
Aggregation :-



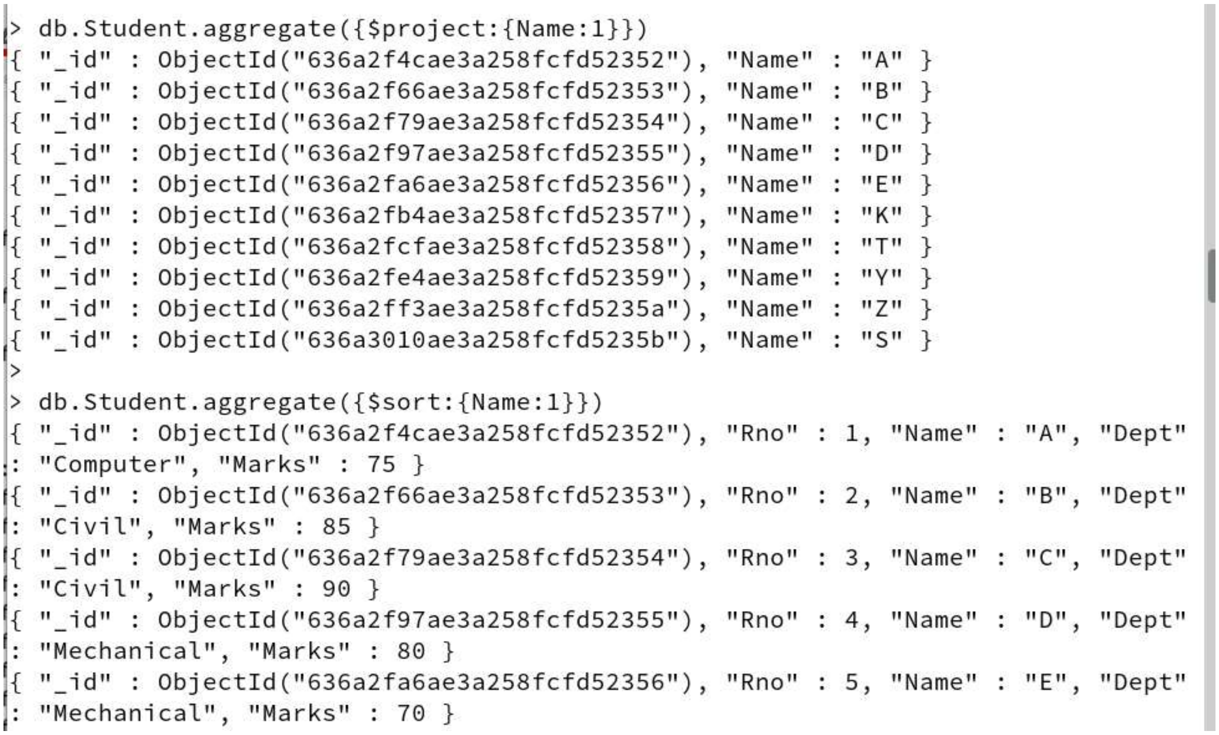
```
---
> show dbs
Practical10  0.000GB
admin        0.000GB
config       0.000GB
local        0.000GB
mydb         0.000GB
> use Practical_11
switched to db Practical_11
> db.createCollection("Student")
{ "ok" : 1 }
> db.Student.find()
> db.Student.insert({Rno:1,Name:"A",Dept:"Computer",Marks:75})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:2,Name:"B",Dept:"Civil",Marks:85})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:3,Name:"C",Dept:"Civil",Marks:90})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:4,Name:"D",Dept:"Mechanical",Marks:80})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:5,Name:"E",Dept:"Mechanical",Marks:70})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:6,Name:"K",Dept:"Mechanical",Marks:65})
WriteResult({ "nInserted" : 1 })
```



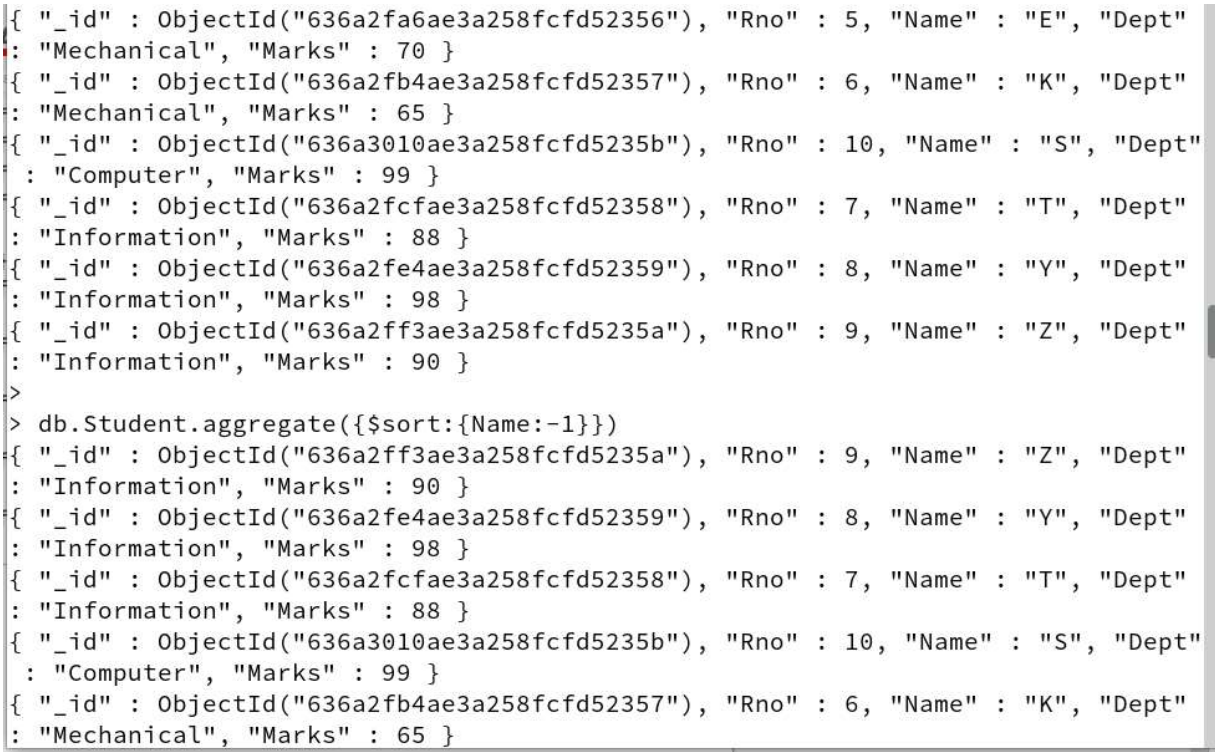
```
> db.Student.insert({Rno:7,Name:"T",Dept:"Information",Marks:88})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:8,Name:"Y",Dept:"Information",Marks:98})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:9,Name:"Z",Dept:"Information",Marks:90})
WriteResult({ "nInserted" : 1 })
> db.Student.insert({Rno:10,Name:"S",Dept:"Computer",Marks:99})
WriteResult({ "nInserted" : 1 })
> db.Student.find()
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5, "Name" : "E", "Dept"
: "Mechanical", "Marks" : 70 }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Rno" : 6, "Name" : "K", "Dept"
: "Mechanical", "Marks" : 65 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7, "Name" : "T", "Dept"
: "Information", "Marks" : 88 }
```



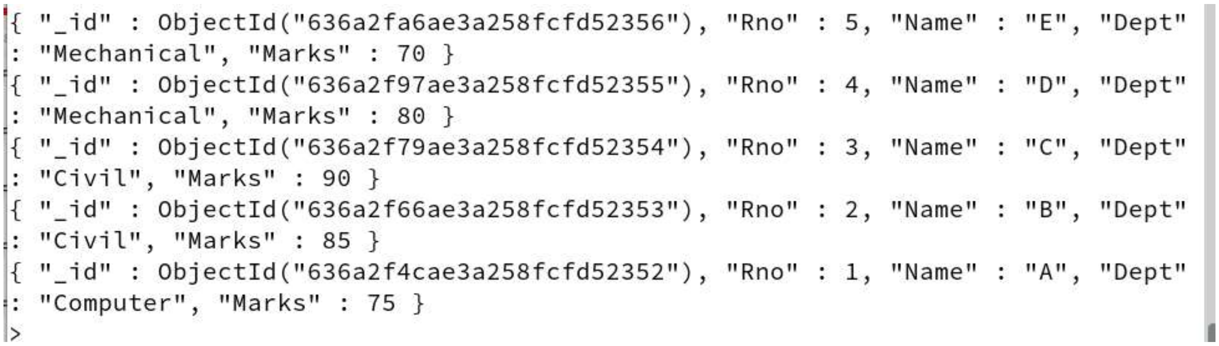
```
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8, "Name" : "Y", "Dept"
: "Information", "Marks" : 98 }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9, "Name" : "Z", "Dept"
: "Information", "Marks" : 90 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
>
>
> db.Student.aggregate({$project:{Rno:1}})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4 }
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5 }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Rno" : 6 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7 }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8 }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10 }
```



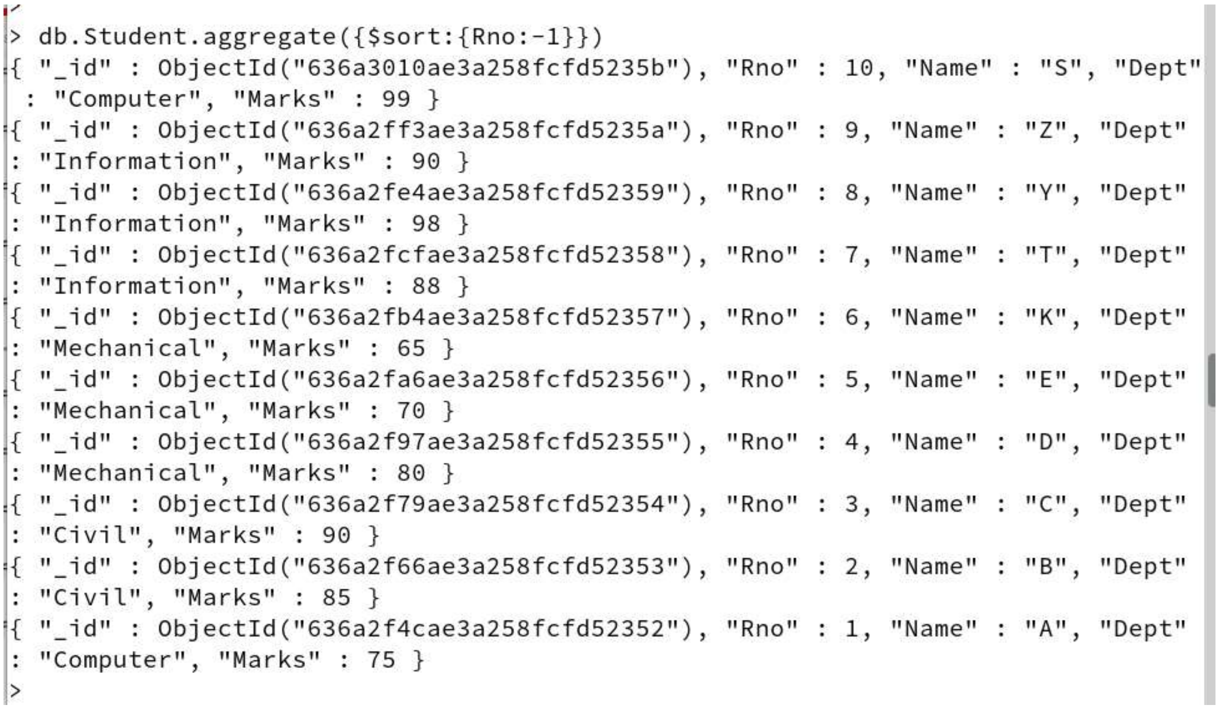
```
> db.Student.aggregate({$project:{Name:1}})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Name" : "A" }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Name" : "B" }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Name" : "C" }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Name" : "D" }
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Name" : "E" }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Name" : "K" }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Name" : "T" }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Name" : "Y" }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Name" : "Z" }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Name" : "S" }
>
> db.Student.aggregate({$sort:{Name:1}})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5, "Name" : "E", "Dept"
: "Mechanical", "Marks" : 70 }
```



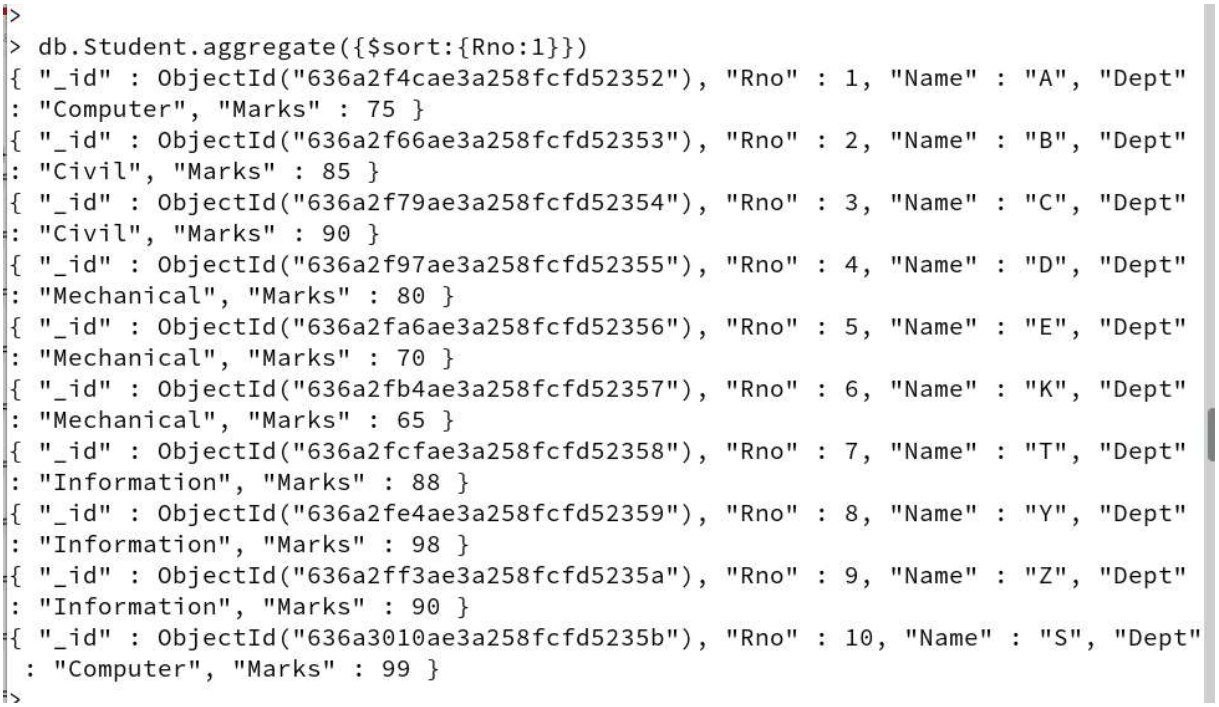
```
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5, "Name" : "E", "Dept"
: "Mechanical", "Marks" : 70 }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Rno" : 6, "Name" : "K", "Dept"
: "Mechanical", "Marks" : 65 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7, "Name" : "T", "Dept"
: "Information", "Marks" : 88 }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8, "Name" : "Y", "Dept"
: "Information", "Marks" : 98 }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9, "Name" : "Z", "Dept"
: "Information", "Marks" : 90 }
>
> db.Student.aggregate({$sort:{Name:-1}})
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9, "Name" : "Z", "Dept"
: "Information", "Marks" : 90 }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8, "Name" : "Y", "Dept"
: "Information", "Marks" : 98 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7, "Name" : "T", "Dept"
: "Information", "Marks" : 88 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Rno" : 6, "Name" : "K", "Dept"
: "Mechanical", "Marks" : 65 }
```



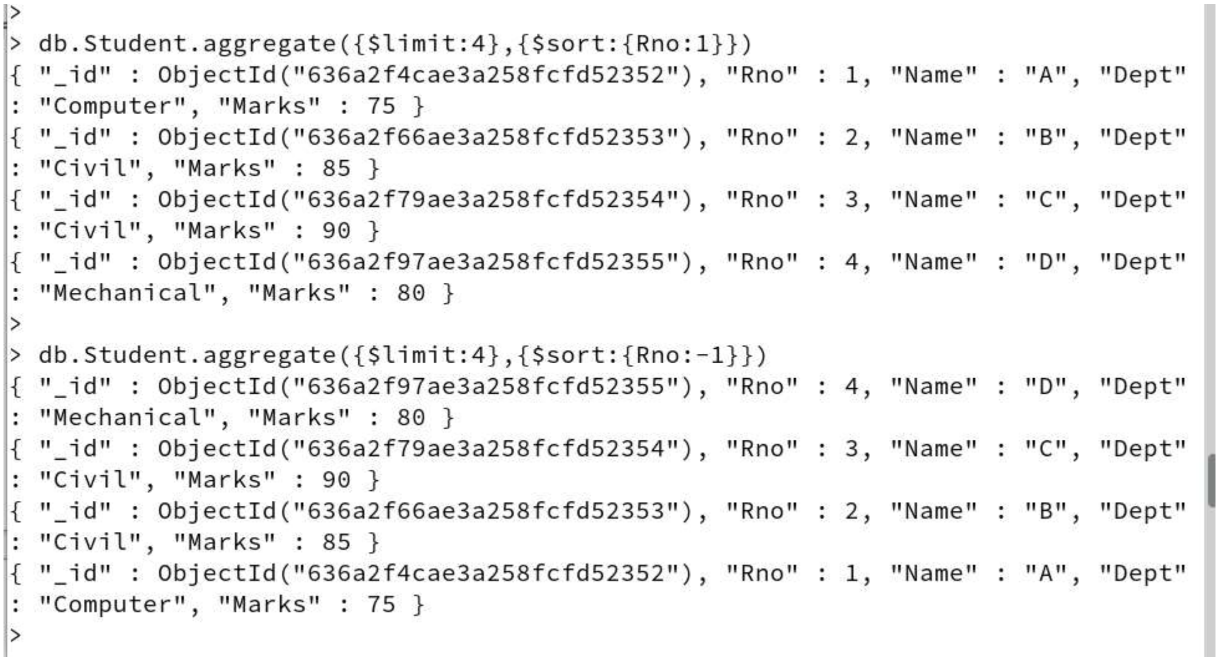
```
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5, "Name" : "E", "Dept"
: "Mechanical", "Marks" : 70 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
>
```



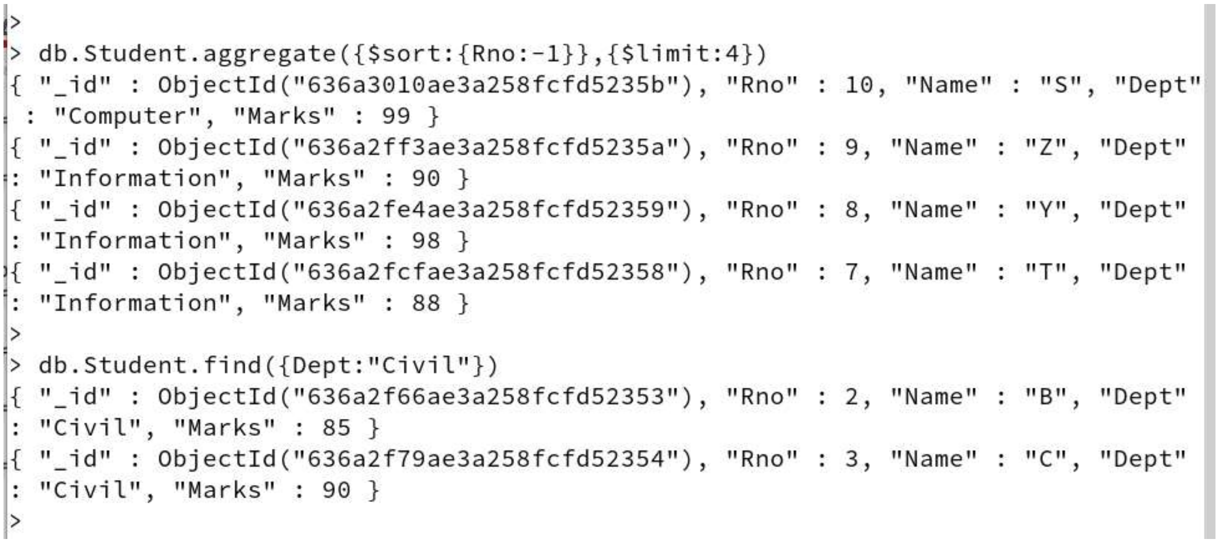
```
> db.Student.aggregate({$sort:{Rno:-1}})
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9, "Name" : "Z", "Dept"
: "Information", "Marks" : 90 }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8, "Name" : "Y", "Dept"
: "Information", "Marks" : 98 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7, "Name" : "T", "Dept"
: "Information", "Marks" : 88 }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Rno" : 6, "Name" : "K", "Dept"
: "Mechanical", "Marks" : 65 }
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5, "Name" : "E", "Dept"
: "Mechanical", "Marks" : 70 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
>
```



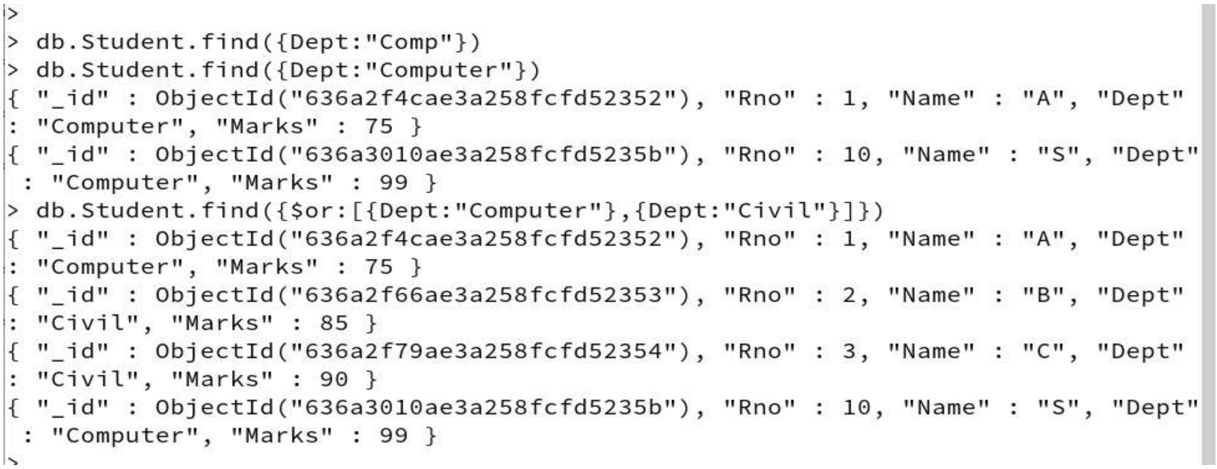
```
>
> db.Student.aggregate({$sort:{Rno:1}})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
{ "_id" : ObjectId("636a2fa6ae3a258fcfd52356"), "Rno" : 5, "Name" : "E", "Dept"
: "Mechanical", "Marks" : 70 }
{ "_id" : ObjectId("636a2fb4ae3a258fcfd52357"), "Rno" : 6, "Name" : "K", "Dept"
: "Mechanical", "Marks" : 65 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7, "Name" : "T", "Dept"
: "Information", "Marks" : 88 }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8, "Name" : "Y", "Dept"
: "Information", "Marks" : 98 }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9, "Name" : "Z", "Dept"
: "Information", "Marks" : 90 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
>
```



```
>
> db.Student.aggregate({$limit:4},{$sort:{Rno:1}})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
>
> db.Student.aggregate({$limit:4},{$sort:{Rno:-1}})
{ "_id" : ObjectId("636a2f97ae3a258fcfd52355"), "Rno" : 4, "Name" : "D", "Dept"
: "Mechanical", "Marks" : 80 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
>
```

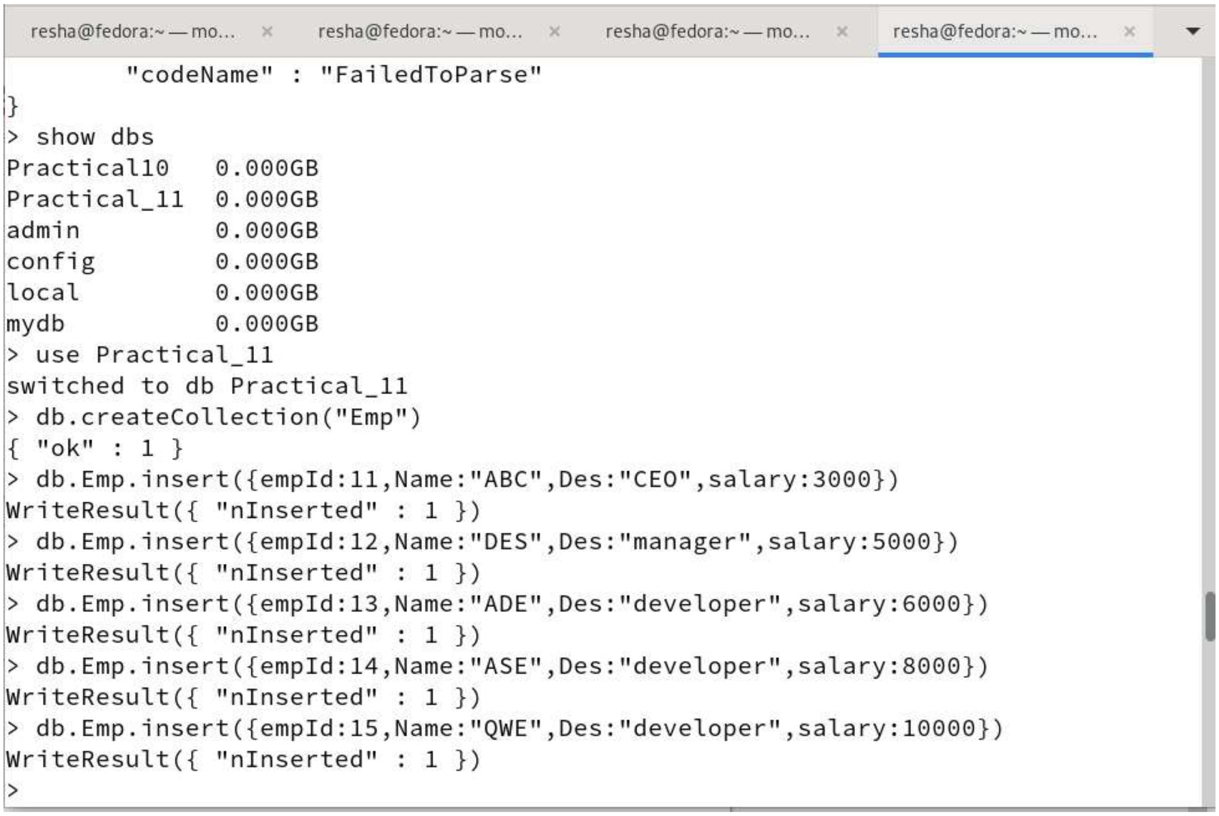


```
>
> db.Student.aggregate({$sort:{Rno:-1}},{$limit:4})
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
{ "_id" : ObjectId("636a2ff3ae3a258fcfd5235a"), "Rno" : 9, "Name" : "Z", "Dept"
: "Information", "Marks" : 90 }
{ "_id" : ObjectId("636a2fe4ae3a258fcfd52359"), "Rno" : 8, "Name" : "Y", "Dept"
: "Information", "Marks" : 98 }
{ "_id" : ObjectId("636a2fcfae3a258fcfd52358"), "Rno" : 7, "Name" : "T", "Dept"
: "Information", "Marks" : 88 }
>
> db.Student.find({Dept:"Civil"})
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
>
```

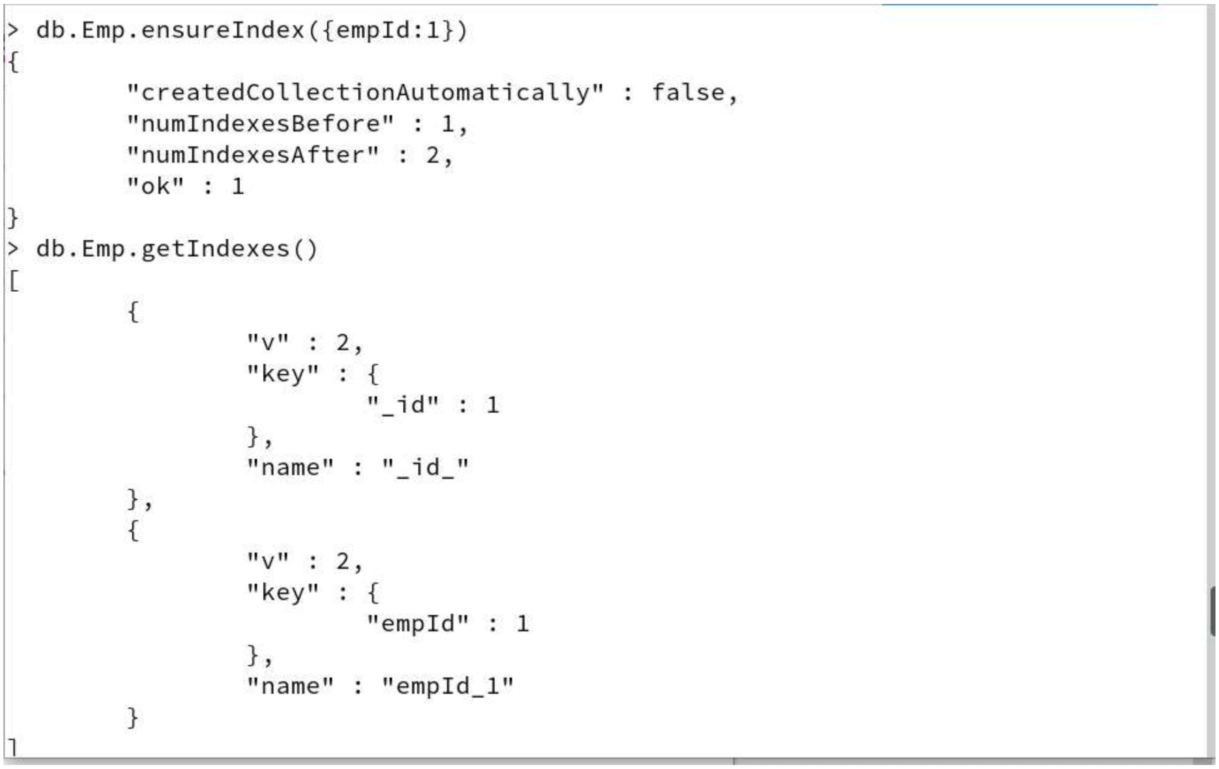


```
>
> db.Student.find({Dept:"Comp"})
> db.Student.find({Dept:"Computer"})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
> db.Student.find({$or:[{Dept:"Computer"},{Dept:"Civil"}]})
{ "_id" : ObjectId("636a2f4cae3a258fcfd52352"), "Rno" : 1, "Name" : "A", "Dept"
: "Computer", "Marks" : 75 }
{ "_id" : ObjectId("636a2f66ae3a258fcfd52353"), "Rno" : 2, "Name" : "B", "Dept"
: "Civil", "Marks" : 85 }
{ "_id" : ObjectId("636a2f79ae3a258fcfd52354"), "Rno" : 3, "Name" : "C", "Dept"
: "Civil", "Marks" : 90 }
{ "_id" : ObjectId("636a3010ae3a258fcfd5235b"), "Rno" : 10, "Name" : "S", "Dept"
 : "Computer", "Marks" : 99 }
>
```

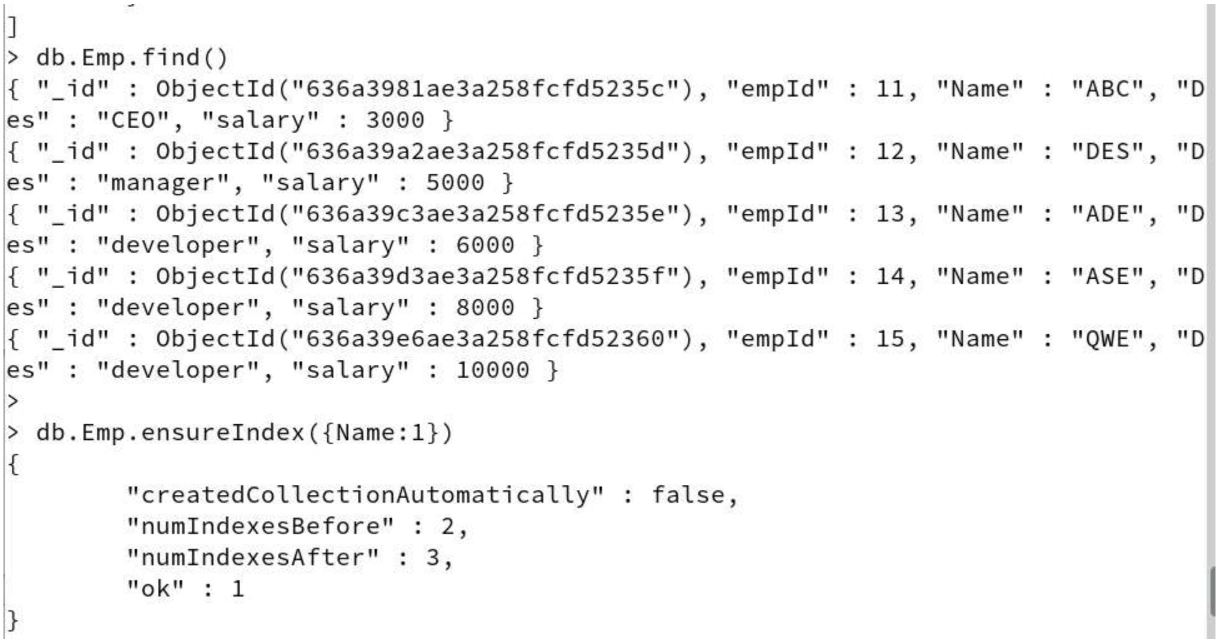
Indexing :-



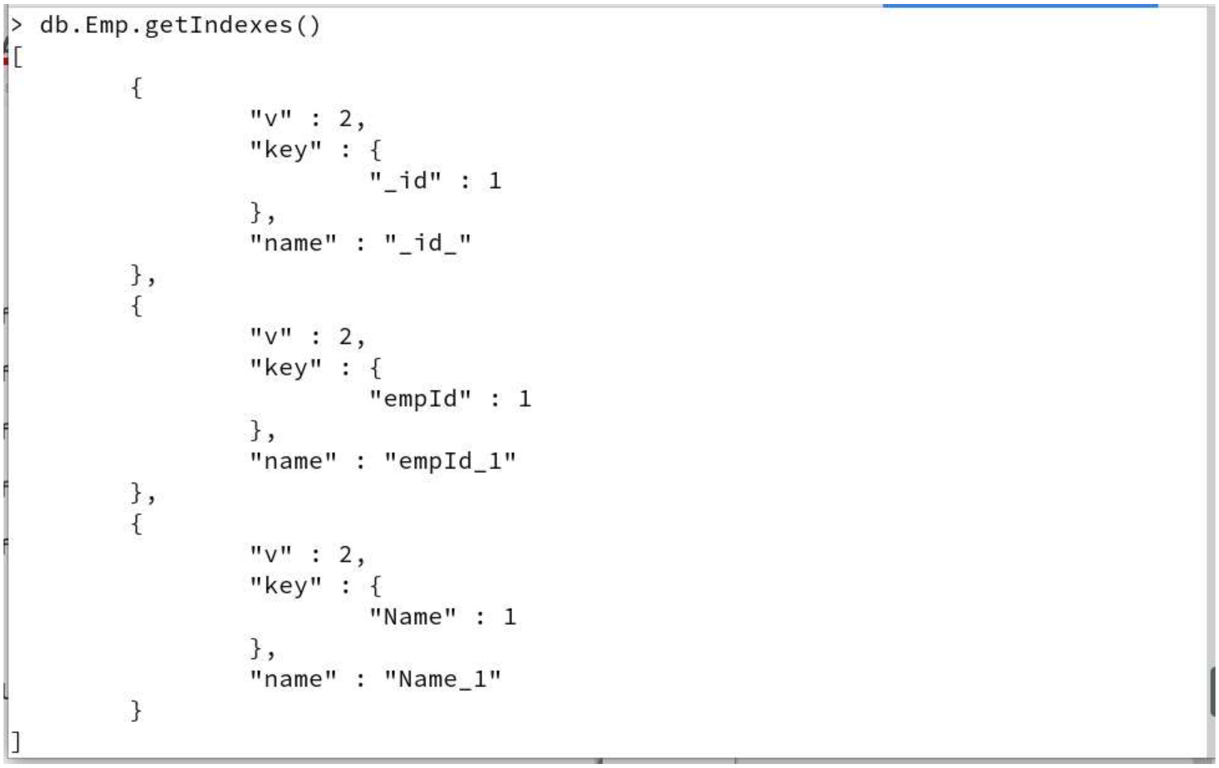
```
        "codeName" : "FailedToParse"
}
> show dbs
Practical10   0.000GB
Practical_11  0.000GB
admin         0.000GB
config        0.000GB
local         0.000GB
mydb          0.000GB
> use Practical_11
switched to db Practical_11
> db.createCollection("Emp")
{ "ok" : 1 }
> db.Emp.insert({empId:11,Name:"ABC",Des:"CEO",salary:3000})
WriteResult({ "nInserted" : 1 })
> db.Emp.insert({empId:12,Name:"DES",Des:"manager",salary:5000})
WriteResult({ "nInserted" : 1 })
> db.Emp.insert({empId:13,Name:"ADE",Des:"developer",salary:6000})
WriteResult({ "nInserted" : 1 })
> db.Emp.insert({empId:14,Name:"ASE",Des:"developer",salary:8000})
WriteResult({ "nInserted" : 1 })
> db.Emp.insert({empId:15,Name:"QWE",Des:"developer",salary:10000})
WriteResult({ "nInserted" : 1 })
>
```



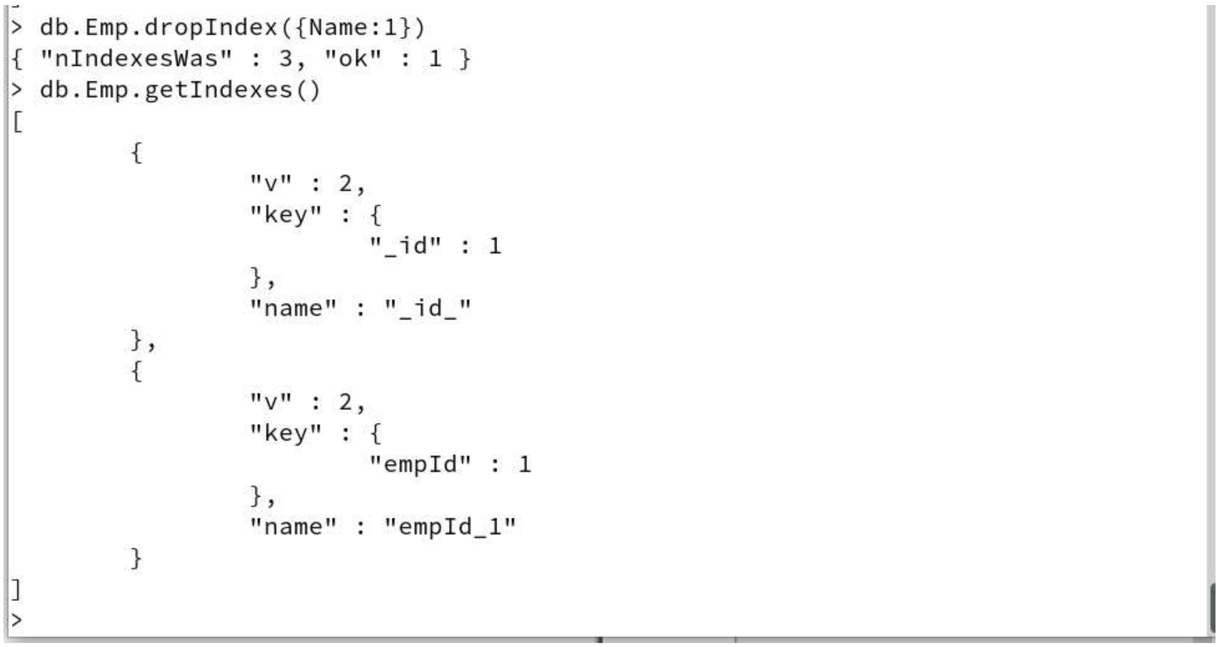
```
> db.Emp.ensureIndex({empId:1})
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 1,
        "numIndexesAfter" : 2,
        "ok" : 1
}
> db.Emp.getIndexes()
[
        {
                "v" : 2,
                "key" : {
                        "_id" : 1
                },
                "name" : "_id_"
        },
        {
                "v" : 2,
                "key" : {
                        "empId" : 1
                },
                "name" : "empId_1"
        }
]
```



```
]
> db.Emp.find()
{ "_id" : ObjectId("636a3981ae3a258fcfd5235c"), "empId" : 11, "Name" : "ABC", "D
es" : "CEO", "salary" : 3000 }
{ "_id" : ObjectId("636a39a2ae3a258fcfd5235d"), "empId" : 12, "Name" : "DES", "D
es" : "manager", "salary" : 5000 }
{ "_id" : ObjectId("636a39c3ae3a258fcfd5235e"), "empId" : 13, "Name" : "ADE", "D
es" : "developer", "salary" : 6000 }
{ "_id" : ObjectId("636a39d3ae3a258fcfd5235f"), "empId" : 14, "Name" : "ASE", "D
es" : "developer", "salary" : 8000 }
{ "_id" : ObjectId("636a39e6ae3a258fcfd52360"), "empId" : 15, "Name" : "QWE", "D
es" : "developer", "salary" : 10000 }
>
> db.Emp.ensureIndex({Name:1})
{
        "createdCollectionAutomatically" : false,
        "numIndexesBefore" : 2,
        "numIndexesAfter" : 3,
        "ok" : 1
}
```



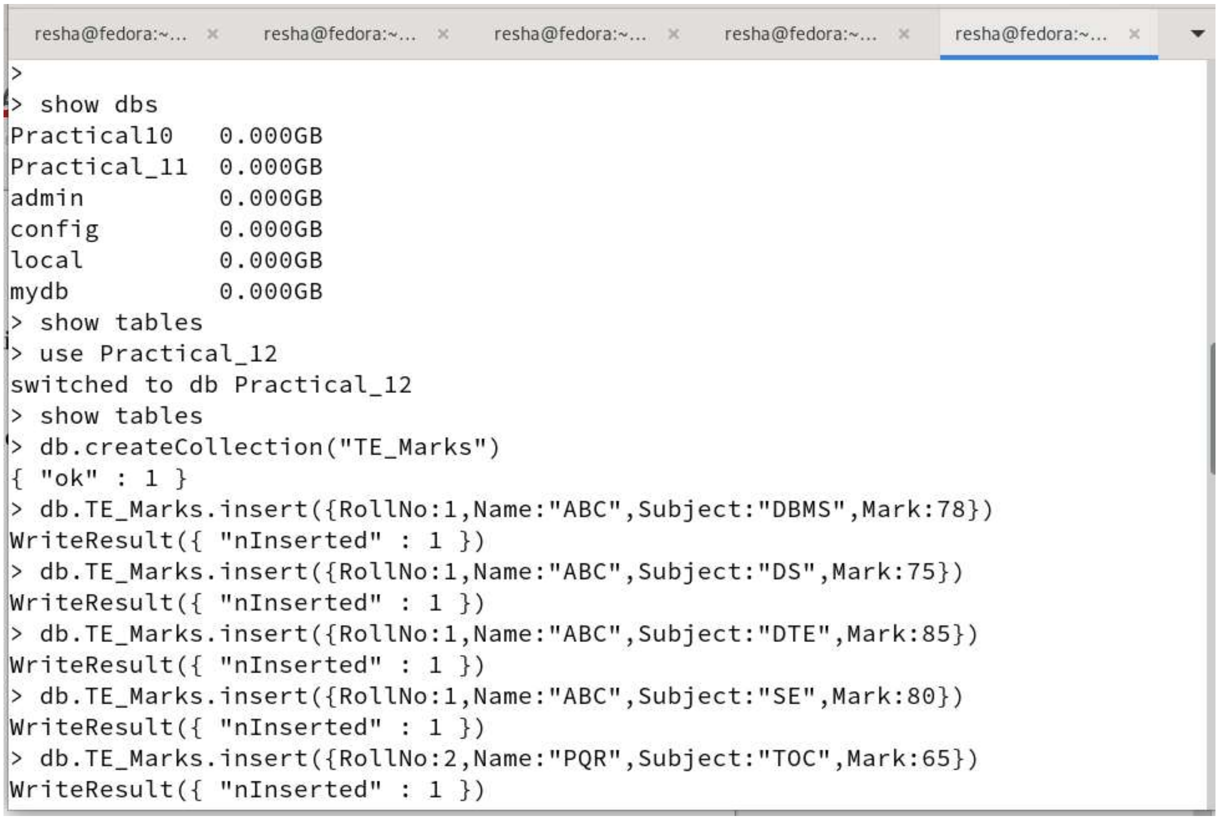
```
> db.Emp.getIndexes()
[
        {
                "v" : 2,
                "key" : {
                        "_id" : 1
                },
                "name" : "_id_"
        },
        {
                "v" : 2,
                "key" : {
                        "empId" : 1
                },
                "name" : "empId_1"
        },
        {
                "v" : 2,
                "key" : {
                        "Name" : 1
                },
                "name" : "Name_1"
        }
]
```



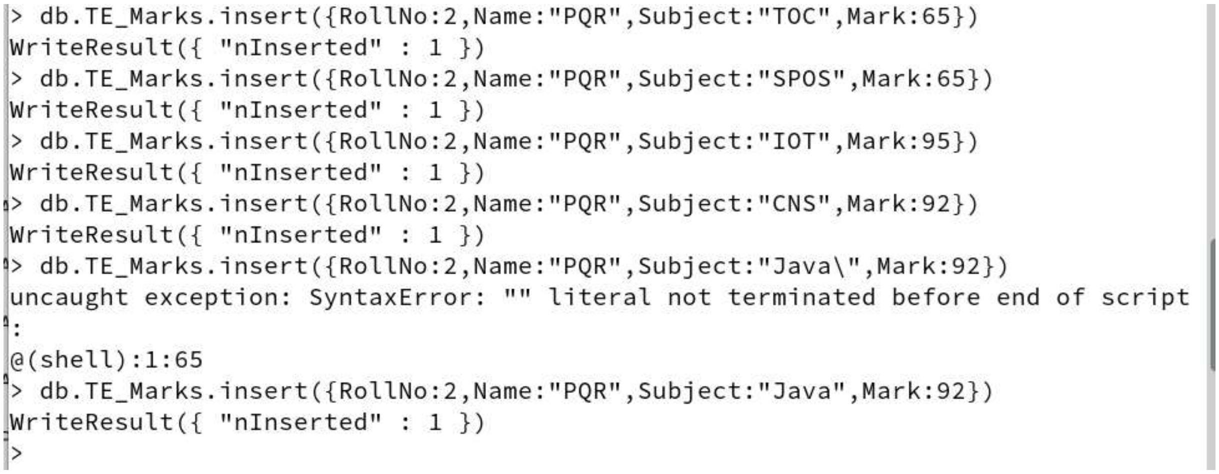
```
> db.Emp.dropIndex({Name:1})
{ "nIndexesWas" : 3, "ok" : 1 }
> db.Emp.getIndexes()
[
        {
                "v" : 2,
                "key" : {
                        "_id" : 1
                },
                "name" : "_id_"
        },
        {
                "v" : 2,
                "key" : {
                        "empId" : 1
                },
                "name" : "empId_1"
        }
]
>
```

Practical No :- 12

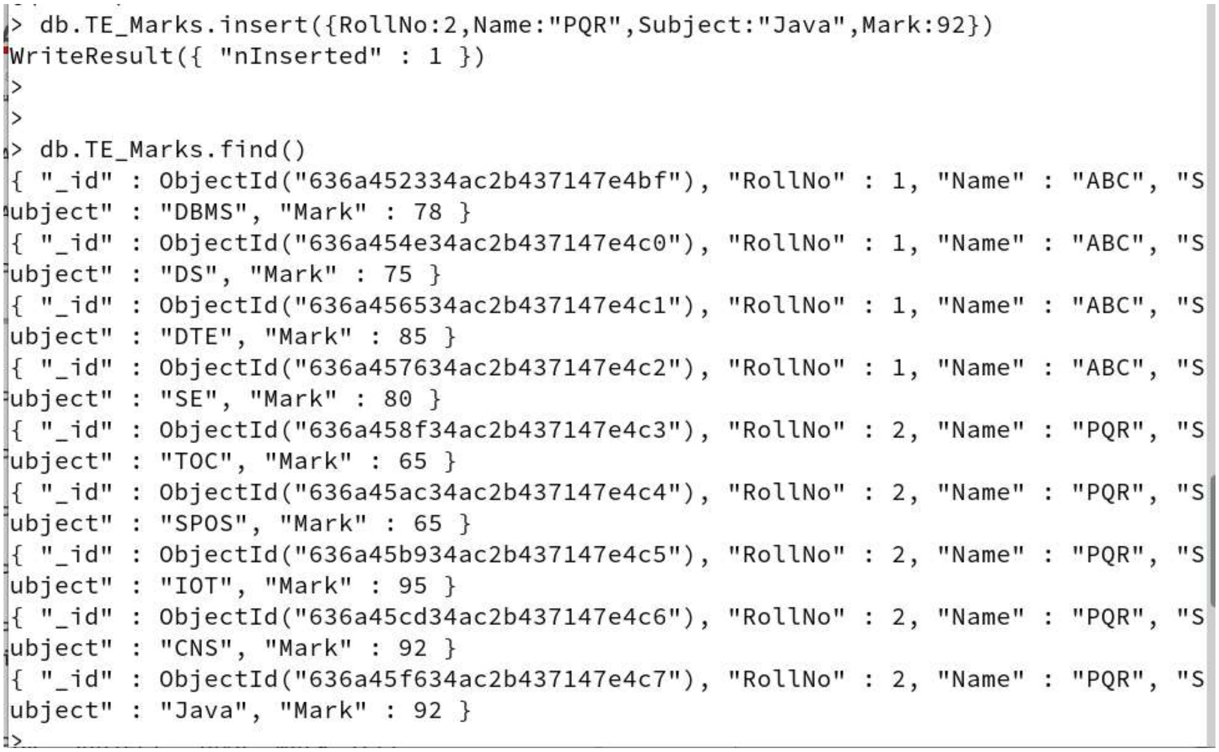
MongoDB – Map-reduces operations:
Implement Map reduces operation with suitable example using MongoDB.



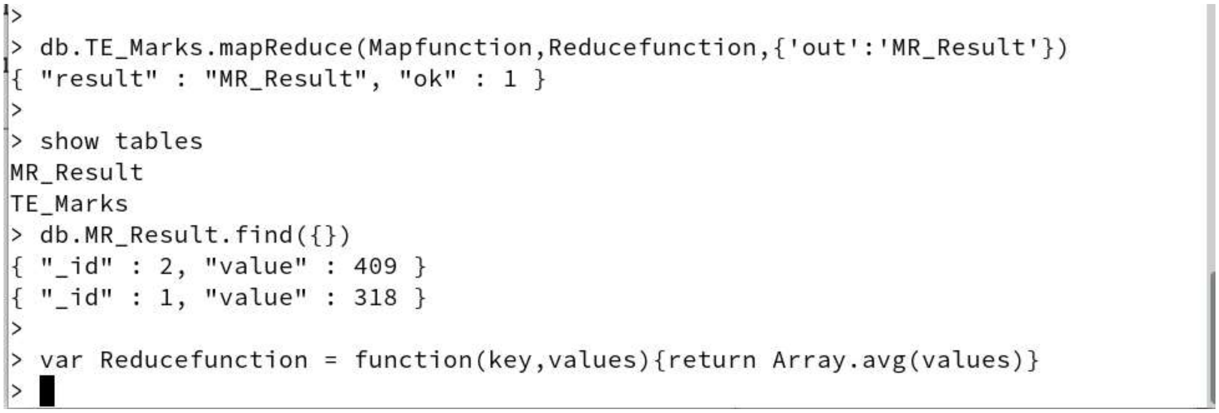
```
>
> show dbs
Practical10   0.000GB
Practical_11  0.000GB
admin         0.000GB
config        0.000GB
local         0.000GB
mydb          0.000GB
> show tables
> use Practical_12
switched to db Practical_12
> show tables
> db.createCollection("TE_Marks")
{ "ok" : 1 }
> db.TE_Marks.insert({RollNo:1,Name:"ABC",Subject:"DBMS",Mark:78})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:1,Name:"ABC",Subject:"DS",Mark:75})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:1,Name:"ABC",Subject:"DTE",Mark:85})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:1,Name:"ABC",Subject:"SE",Mark:80})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"TOC",Mark:65})
WriteResult({ "nInserted" : 1 })
```



```
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"TOC",Mark:65})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"SPOS",Mark:65})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"IOT",Mark:95})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"CNS",Mark:92})
WriteResult({ "nInserted" : 1 })
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"Java\",Mark:92})
uncaught exception: SyntaxError: "" literal not terminated before end of script
:
@(shell):1:65
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"Java",Mark:92})
WriteResult({ "nInserted" : 1 })
>
```



```
> db.TE_Marks.insert({RollNo:2,Name:"PQR",Subject:"Java",Mark:92})
WriteResult({ "nInserted" : 1 })
>
>
> db.TE_Marks.find()
{ "_id" : ObjectId("636a452334ac2b437147e4bf"), "RollNo" : 1, "Name" : "ABC", "S
ubject" : "DBMS", "Mark" : 78 }
{ "_id" : ObjectId("636a454e34ac2b437147e4c0"), "RollNo" : 1, "Name" : "ABC", "S
ubject" : "DS", "Mark" : 75 }
{ "_id" : ObjectId("636a456534ac2b437147e4c1"), "RollNo" : 1, "Name" : "ABC", "S
ubject" : "DTE", "Mark" : 85 }
{ "_id" : ObjectId("636a457634ac2b437147e4c2"), "RollNo" : 1, "Name" : "ABC", "S
ubject" : "SE", "Mark" : 80 }
{ "_id" : ObjectId("636a458f34ac2b437147e4c3"), "RollNo" : 2, "Name" : "PQR", "S
ubject" : "TOC", "Mark" : 65 }
{ "_id" : ObjectId("636a45ac34ac2b437147e4c4"), "RollNo" : 2, "Name" : "PQR", "S
ubject" : "SPOS", "Mark" : 65 }
{ "_id" : ObjectId("636a45b934ac2b437147e4c5"), "RollNo" : 2, "Name" : "PQR", "S
ubject" : "IOT", "Mark" : 95 }
{ "_id" : ObjectId("636a45cd34ac2b437147e4c6"), "RollNo" : 2, "Name" : "PQR", "S
ubject" : "CNS", "Mark" : 92 }
{ "_id" : ObjectId("636a45f634ac2b437147e4c7"), "RollNo" : 2, "Name" : "PQR", "S
ubject" : "Java", "Mark" : 92 }
>
```



```
>
> db.TE_Marks.mapReduce(Mapfunction,Reducefunction,{'out':'MR_Result'})
{ "result" : "MR_Result", "ok" : 1 }
>
> show tables
MR_Result
TE_Marks
> db.MR_Result.find({})
{ "_id" : 2, "value" : 409 }
{ "_id" : 1, "value" : 318 }
>
> var Reducefunction = function(key,values){return Array.avg(values)}
>
```

Practical No :- 13

JAVA CODE TO CONNECT TO MONGODB DATABASE

```
import com.mongodb.client.MongoDatabase;
import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
public class ConnectToDB {

  public static void main( String args[] ) {

    // Creating a Mongo client
    MongoClient mongo = new MongoClient( "localhost" , 27017 );

    // Creating Credentials
    MongoCredential credential;
    credential = MongoCredential.createCredential("sampleUser", "myDb",
      "password".toCharArray());
    System.out.println("Connected to the database successfully");

    // Accessing the database
    MongoDatabase database = mongo.getDatabase("myDb");
    System.out.println("Credentials ::"+ credential);
  }
}
```

OUTPUT:

```
Connected to the database successfully
Credentials ::MongoCredential{
  mechanism = null,
  userName = 'sampleUser',
  source = 'myDb',
  password = <hidden>,
  mechanismProperties = {}
}
```

JAVA CODE TO CREATE COLLECTION

```
import com.mongodb.client.MongoDatabase;
import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
public class CreatingCollection {

  public static void main( String args[] ) {

    // Creating a Mongo client
    MongoClient mongo = new MongoClient( "localhost" , 27017 );

    // Creating Credentials
    MongoCredential credential;
    credential = MongoCredential.createCredential("sampleUser", "myDb",
      "password".toCharArray());
    System.out.println("Connected to the database successfully");

    //Accessing the database
    MongoDatabase database = mongo.getDatabase("myDb");

    //Creating a collection
    database.createCollection("sampleCollection");
    System.out.println("Collection created successfully");
  }
}
```

JAVA CODE FOR GETTING/SELECTING A COLLECTION

```
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import org.bson.Document;
import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
public class selectingCollection {

  public static void main( String args[] ) {

    // Creating a Mongo client
    MongoClient mongo = new MongoClient( "localhost" , 27017 );

    // Creating Credentials
    MongoCredential credential;
    credential = MongoCredential.createCredential("sampleUser", "myDb",
      "password".toCharArray());
    System.out.println("Connected to the database successfully");

    // Accessing the database
    MongoDatabase database = mongo.getDatabase("myDb");

    // Creating a collection
    System.out.println("Collection created successfully");
    // Retrieving a collection
    MongoCollection<Document> collection = database.getCollection("myCollection");
    System.out.println("Collection myCollection selected successfully");
  }
}
```

JAVA CODE FOR INSERTING A DOCUMENT

```
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import org.bson.Document;
import com.mongodb.MongoClient;
public class InsertingDocument {
        public static void main( String args[] ) {

        // Creating a Mongo client
        MongoClient mongo = new MongoClient( "localhost" , 27017 );

        // Accessing the database
        MongoDatabase database = mongo.getDatabase("myDb");

        // Creating a collection
        database.createCollection("sampleCollection");
        System.out.println("Collection created successfully");

        // Retrieving a collection
        MongoCollection<Document> collection =
database.getCollection("sampleCollection");
        System.out.println("Collection sampleCollection selected successfully");
        Document document = new Document("title", "MongoDB")
        .append("description", "database")
        .append("likes", 100)
        .append("url", "http://www.tutorialspoint.com/mongodb/")
        .append("by", "tutorials point");

        //Inserting document into the collection
        collection.insertOne(document);
        System.out.println("Document inserted successfully");
}
```

OUTPUT:

```
Connected to the database successfully
Collection sampleCollection selected successfully
Document inserted successfully
```