```
omii@omii:~$ sudo service mongod start
[sudo] password for omii:
omii@omii:~$ mongosh
Current Mongosh Log ID: 652530563ad87d71dd3e1e54
Connecting to:
mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName
=mongosh+2.0.1
Using MongoDB: 7.0.2
Using Mongosh: 2.0.1

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

------
   The server generated these startup warnings when booting
   2023-10-11T04:15:14.689+05:30: Using the XFS filesystem is strongly recommended with the
WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
   2023-10-11T04:15:15.022+05:30: Access control is not enabled for the database. Read and write
access to data and configuration is unrestricted
   2023-10-11T04:15:15.022+05:30: vm.max_map_count is too low
------

test>use Stud
switched to db Stud
Stud> db.createCollection("Stud_info");
{ ok: 1 }
Stud> db.Stud_info.insertOne({Name:"Anjali",Id:"01",Subject:"DBMS",Marks:"75"});
{
  acknowledged: true,
  insertedId: ObjectId("652530563ad87d71dd3e1e55")
Stud> db.Stud_info.insertOne({Name:"Om",Id:"02",Subject:"DBMS",Marks:"80"});
{
  acknowledged: true,
  insertedId: ObjectId("6525307b3ad87d71dd3e1e56")
}
Stud> db.Stud_info.insertOne({Name:"Mayur",Id:"03",Subject:"DBMS",Marks:"69"});
{
  acknowledged: true,
  insertedId: ObjectId("652530b23ad87d71dd8e1e57")
}
Stud> db.Stud_info.insertOne({Name:"Dipali",Id:"04",Subject:"DBMS",Marks:"45"});
{
  acknowledged: true,
  insertedId: ObjectId("652530c43ad87d71dd8e1e58")
}
Stud> db.Stud_info.insertOne({Name:"Rohini",Id:"05",Subject:"DBMS",Marks:"39"});
{
  acknowledged: true,
  insertedId: ObjectId("652530d73ad87d71dd8e1e59")
}

Stud> var mapFunction = function () {
```

```
...    var category;
...    if (this.Marks > 70) {
...        category = "High scores";
...    } else if (this.Marks > 40) {
...        category = "Average scores";
...    } else {
...        category = "Failed";
...    }
...    emit(category, 1);
... };

Stud> db.Stud_info.mapReduce(mapFunction, reduceFunction, { out: "score_categories" });
{ result: 'score_categories', ok: 1 }

Stud> db.score_categories.find();
[
  { _id: 'Failed', value: 1 },
  { _id: 'Average scores', value: 2 },
  { _id: 'High scores', value: 2 }
]
```