

Lab – 6

CE004

20CEUOD004

MongoDB quick tour

```
File Edit View Search Terminal Help
db.setVerboseShell(flag) display extra information in shell output
db.shutdownServer()
db.stats()
db.version() current version of the server
> mongod
2022-09-02T15:42:31.850+0530 E QUERY [thread1] ReferenceError: mongod is not defined :
@ (shell):1:1
> mongod&
...
...
> mongod
2022-09-02T15:43:32.775+0530 E QUERY [thread1] ReferenceError: mongod is not defined :
@ (shell):1:1
> mongod
2022-09-02T15:43:55.367+0530 E QUERY [thread1] ReferenceError: mongod is not defined :
@ (shell):1:1
> show dbs
admin 0.000GB
local 0.000GB
> show dbs
admin 0.000GB
local 0.000GB
> db
test
> use demodb
switched to db demodb
> db.createCollection('democollection');
{ "ok" : 1 }
> db.democollection.save({__id:1,name:"rahil"});
WriteResult({ "nInserted" : 1 })
> db.democollection.find({});
{ "_id" : ObjectId("6311d80ca1b6c68172f3ab64"), "__id" : 1, "name" : "rahil" }
> sho dbs
2022-09-02T15:51:28.615+0530 E QUERY [thread1] SyntaxError: missing ; before statement @ (shell):1:4
> sho dbs;
```

Windows taskbar: MongoDB quick tour - Google Docs..., hadoop@hadoop-clone:~

```
File Edit View Search Terminal Help
> use demodb
switched to db demodb
> show collections
demoCollection
democollection
> db.createCollection("students")
{ "ok" : 1 }
> show collections
demoCollection
democollection
students
> db.students.insert({ _id:1,studentRollNo:'s101',studentName:'rahil',grade:'VIII',hobbies:'cricket'})
WriteResult({ "nInserted" : 1 })
> db.democollection.find({});
{ "_id" : ObjectId("6311d80ca1b6c68172f3ab64"), "__id" : 1, "name" : "rahil" }
> db.students.find({});
{ "_id" : ObjectId("6311da06a1b6c68172f3ab65"), "__id" : 1, "studentRollNo" : "s101", "studentName" : "rahil", "grade" : "VIII", "hobbies" : "cricket" }
> db.students.find({}).pretty();
{
  "_id" : ObjectId("6311da06a1b6c68172f3ab65"),
  "__id" : 1,
  "studentRollNo" : "s101",
  "studentName" : "rahil",
  "grade" : "VIII",
  "hobbies" : "cricket"
}
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
2022-09-02T15:58:44.651+0530 E QUERY [thread1] SyntaxError: expected expression, got ')' @(shell):1:72
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
2022-09-02T16:00:33.186+0530 E QUERY [thread1] SyntaxError: expected expression, got ')' @(shell):1:72
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
...
...
>
```

```
MongoDB quick tour - Google Docs... hadoop@hadoop-clone:~ Pictures 1 / 4
File Edit View Search Terminal Help
democollection
> db.createCollection("students")
{ "ok" : 1 }
> show collections
demoCollection
democollection
students
> db.students.insert({ _id:1,studentRollNo:'s101',studentName:'rahil',grade:'VIII',hobbies:'cricket'})
WriteResult({ "nInserted" : 1 })
> db.democollection.find({});
{ "_id" : ObjectId("6311d80ca1b6c68172f3ab64"), "__id" : 1, "name" : "rahil" }
> db.students.find({});
{ "_id" : ObjectId("6311da06a1b6c68172f3ab65"), "__id" : 1, "studentRollNo" : "s101", "studentName" : "rahil", "grade" : "VIII", "hobbies" : "cricket" }
> db.students.find({}).pretty();
{
  "_id" : ObjectId("6311da06a1b6c68172f3ab65"),
  "__id" : 1,
  "studentRollNo" : "s101",
  "studentName" : "rahil",
  "grade" : "VIII",
  "hobbies" : "cricket"
}
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
2022-09-02T15:58:44.651+0530 E QUERY [thread1] SyntaxError: expected expression, got ')' @(shell):1:72
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
2022-09-02T16:00:33.186+0530 E QUERY [thread1] SyntaxError: expected expression, got ')' @(shell):1:72
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
...
...
>
>
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}},)
2022-09-02T16:01:41.193+0530 E QUERY [thread1] SyntaxError: expected expression, got ')' @(shell):1:72
> db.students.update({studentRollNo:'s101'},{$set:{hobbies:'Ice hockey'}})
...
...
>
```

Map reduce:

```
> db.bookdb.insertMany([
...   {
...     "id" : 1,
...     "Category" : "Machine Learning",
...     "Bookname" : "Machine Learning for Hackers",
...     "Author" : "Drew Conway",
...     "qty" : 25,
...     "price" : 400,
...     "rol" : 30,
...     "pages" : 350
...   },
...   {
...     "id" : 2,
...     "Category" : "Business Intelligence",
...     "Bookname" : "Fundamentals of Business Analytics",
...     "Author" : "Seema Acharya",
...     "qty" : 55,
...     "price" : 500,
...     "rol" : 30,
...     "pages" : 250
...   }
... ])
{ "acknowledged" : true, "insertedIds" : [ 1, 2 ] }
```

```
> db.bookdb.find({}).pretty();
{
  "_id" : 1,
  "Category" : "Machine Learning",
  "Bookname" : "Machine Learning for Hackers",
  "Author" : "Drew Conway",
  "qty" : 25,
  "price" : 400,
  "rol" : 30,
  "pages" : 350
}
{
  "_id" : 2,
  "Category" : "Business Intelligence",
  "Bookname" : "Fundamentals of Business Analytics",
  "Author" : "Seema Acharya",
  "qty" : 55,
  "price" : 500,
  "rol" : 30,
  "pages" : 250
}
{
  "_id" : 3,
  "Category" : "Analytics",
  "Bookname" : "Competing on Analytics",
  "Author" : "Thomas Davenport",
  "qty" : 8,
  "price" : 150,
  "rol" : 20,
  "pages" : 150
}
{
```

```
File Edit View Search Terminal Help
{
  "id" : 3,
  "Category" : "Analytics",
  "Bookname" : "Competing on Analytics",
  "Author" : "Thomas Davenport",
  "qty" : 8,
  "price" : 150,
  "rol" : 20,
  "pages" : 150
}
{
  "id" : 4,
  "Category" : "Visualization",
  "Bookname" : "Visualizing Data",
  "Author" : "Ben Fry",
  "qty" : 12,
  "price" : 325,
  "rol" : 6,
  "pages" : 450
}
{
  "id" : 5,
  "Category" : "Web Mining",
  "Bookname" : "Learning R",
  "Author" : "Richard Cotton",
  "qty" : 5,
  "price" : 850,
  "rol" : 10,
  "pages" : 120
}
> db
bookdb
> db.bookdb.find({"rol":10})
{ "id" : 5, "Category" : "Web Mining", "Bookname" : "Learning R", "Author" : "Richard Cotton", "qty" : 5, "price" : 850, "rol" : 10, "pages" : 120 }
> []
```

```
File Edit View Search Terminal Help
  "rol" : 30,
  "pages" : 250
}
{
  var map = function(){ if ( this.pages > 300) emit ('Big Books',1); else emit ('Small Books',1); }
>
> var reduce = function(key, values){ return Array.sum(values);}
>
> db.books.mapReduce(map, reduce, {out:"Book_Result",query:{}});
2022-09-02T16:57:37.629+0530 E QUERY [thread1] Error: map reduce failed:{ "ok" : 0, "errmsg" : "ns doesn't exist" } :
getErrorWithCode@src/mongo/shell/utils.js:25:13
DBCollection.prototype.mapReduce@src/mongo/shell/collection.js:1411:1
@(shell):1:1
> db.bookdb.mapReduce(map, reduce, {out:"Book_Result",query:{}});
{
  "result" : "Book_Result",
  "timeMillis" : 529,
  "counts" : {
    "input" : 5,
    "emit" : 5,
    "reduce" : 2,
    "output" : 2
  },
  "ok" : 1
}
> db.Book_Result.find({});
{ "_id" : "Big Books", "value" : 2 }
{ "_id" : "Small Books", "value" : 3 }
> []
```