MongoDB Map Reduce – mapReduce() Example

MongoDB Map Reduce

In this MongoDB Tutorial – **MongoDB Map Reduce**, we shall learn to use mapReduce() function for performing aggregation operations on a MongoDB Collection, with the help of examples.

Syntax of Mongo mapReduce()

Following is the syntax of mapReduce() function that could be used in Mongo Shell

```
>db.collection.mapRe
```

```
>db.collection.mapReduce(
  function() {emit(key,value);}, // map function
  function(key,values) {return reduceFunction}, // reduce function
  { out: collection }
)
```

Example for MongoDB mapReduce()

In this example we shall take school db in which students is a collection and the collection has documents where each document has name of the student, marks he/she scored in a particular subject. We shall apply mapReduce function to accumulate the marks for each student.

Following is the students collection.

```
> db.students.find({});
```

```
> db.students.find({});
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec39"), "name" : "Midhu", "subject" : "science", "marks" : 68 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec3a"), "name" : "Midhu", "subject" : "maths", "marks" : 98 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec3b"), "name" : "Midhu", "subject" : "sports", "marks" : 77 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec3c"), "name" : "Akhil", "subject" : "science", "marks" : 67 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec3d"), "name" : "Akhil", "subject" : "maths", "marks" : 87 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec3e"), "name" : "Akhil", "subject" : "sports", "marks" : 89 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec3f"), "name" : "Anish", "subject" : "science", "marks" : 67 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec40"), "name" : "Anish", "subject" : "maths", "marks" : 78 }
{ "_id" : ObjectId("5a1f9ce431c157f3ec2aec41"), "name" : "Anish", "subject" : "sports", "marks" : 90 }
```

mapReduce() In Mongo Shell

Following is a step by step guide to prepare mapReduce function for the use case in Mongo Shell :

1. ### Prepare Map function.

   Our map function should emit key-value pair. And in this case, name is key and value is marks.

```
var map = function()
```

```
var map = function() {emit(this.name,this.marks);};
```

2. ### Prepare Reduce function.

   Our map function should emit key-value pair. And in this case, name is key and value is marks.

```
var reduce =
```

```
var reduce = function(name,marks) {return Array.sum(marks);};
```

3. ### Prepare mapReduce function.

   Our map function should emit key-value pair. And in this case, name is key and value is marks.

```
db.students.mapReduc
```

```
db.students.mapReduce(
  map,
  reduce,
  { out: "totals" }
);
```

   `out: "totals"` : the output is written to totals collection in the same database.

4. ### Start Mongo Daemon.

   Run the following command in terminal to start mongo daemon.

```
~$ sudo mongod --
```

```
~$ sudo mongod --port 27017 --dbpath /var/lib/mongodb
```

Now Mongo Daemon would be waiting for connections on port 27017.

5. ## Run mapReduce.

Start a Mongo Shell and Run the above commands (in Step 1 to Step 3) in Mongo Shell.

Mongo Shell

```
> var map = function()
{emit(this.name,this.m
```

```
> var map = function() {emit(this.name,this.marks);};
> var reduce = function(name,marks) {return Array.sum(marks);};
> db.students.mapReduce(
... map,
... reduce,
... { out: "totals" }
... );
{
    "result" : "totals",
    "timeMillis" : 599,
    "counts" : {
        "input" : 9,
        "emit" : 9,
        "reduce" : 3,
        "output" : 3
    },
    "ok" : 1
}
> db.totals.find({})
{ "_id" : "Akhil", "value" : 243 }
{ "_id" : "Anish", "value" : 235 }
{ "_id" : "Midhu", "value" : 243 }
```

The value has been accumulated (aggregated) for the key values and the output is written to totals collection.

Using mapReduce Command in Mongo Script

Following is the Mongo Script file that runs mapReduce command and writes result to totals collection

mongo-mapreduce-example.js

```
// equivalent for "use
<db>" command in
```

```
// equivalent for "use <db>" command in mongo shell
db = db.getSiblingDB('school')

db.runCommand( {
    mapReduce: "students",
    map: function() {
        emit( this.name, this.marks );
    },
    reduce: function(name, values) {
        var value = 0;
        for (var index = 0; index < values.length; ++index) {
            value += values[index];
        }
        return value;
    },
    out: {
        replace: "totals"
    }
} )
```

Run the JavaScript file using mongo command

Mongo Script Output

```
~$ mongo mongo-mapreduce-example.js
```

```
~$ mongo mongo-mapreduce-example.js
MongoDB shell version v3.4.10
connecting to: mongodb://127.0.0.1:27017
MongoDB server version: 3.4.10
```

You may check the totals collection

Mongo Shell

```
> db.totals.find({})
{ "_id" : "Akhil", "value"
```

```
> db.totals.find({})
{ "_id" : "Akhil", "value" : 243 }
{ "_id" : "Anish", "value" : 235 }
{ "_id" : "Midhu", "value" : 243 }
```

- *You may choose any property as a key or value from the collection for the mapReduce() function.*

Conclusion

We have learnt how to use MongoDB MapReduce function in Mongo Shell and Mongo Script with examples.

MongoDB Tutorial

- MongoDB Tutorial
- Install MongoDB on Ubuntu
- Start MongoDB Server
- MongoDB Shell
- Check MongoDB Version
- MongoDB Server Port Number - Default Value and How to Change it
- MongoDB Script

Database

- MongoDB Database
- MongoDB Create Database
- MongoDB Delete Database

Collections

- MongoDB Collection
- MongoDB Create Collection
- MongoDB Delete Collection

Documents

- MongoDB Document
- MongoDB Insert Document
- MongoDB Query Documents
- MongoDB Project Fields in Result
- MongoDB Update Document
- MongoDB Delete Document
- MongoDB Limit Documents
- MongoDB Skip Documents
- MongoDB Sort Documents
- MongoDB Setup Replica Set
- MongoDB Locks

MongoDB Concepts

- MongoDB Text Search

MongoDB MapReduce
MongoDB Backup - mongodump

MongoDB Queries

MongoDB Date

MongoDB Queries

MongoDB Date()

MongoDB Integration

MongoDB Java

Connect to MongoDB from Java

MongoDB Python

Connect to MongoDB from Python

MongoDB Kotlin

Connect to MongoDB from Kotlin

MongoDB Node.js

Node.js MongoDB
Node.js MongoDB Connection
Node.js MongoDB Create Database
Node.js MongoDB Drop Database
Node.js MongoDB Create Collection
Node.js MongoDB Delete Collection
Node.js MongoDB Insert Documents
MongoError: failed to connect to server

MongoDB Others

MongoDB Interview Questions
Uninstall MongoDB from Ubuntu

Useful Resources

How to Learn Programming