Zeal College of Engineering and Research

Subject: DBMSL

Name: Jenil Girish Rathod   Class: TE

Div: B   Batch: B1

Roll No: T212009

Group B: Practical No: 12

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

Code:
```
test> use map
switched to db map
map> db.createCollection("books")
{ ok: 1 }
map> db.books.insertOne({'name':'Harry Potter','pages':'100'})
{
 acknowledged: true,
 insertedId: ObjectId('670a1607d11af492e486b01d')
}
map> db.books.insertOne({'name':'Percy','pages':'200'})
{
 acknowledged: true,
 insertedId: ObjectId('670a1619d11af492e486b01e')
}
map> db.books.insertOne({'name':'X-men','pages':'300'})
{
 acknowledged: true,
 insertedId: ObjectId('670a1630d11af492e486b01f')
}
map> db.books.insertOne({'name':'Catwoman','pages':'350'})
{
 acknowledged: true,
 insertedId: ObjectId('670a164dd11af492e486b020')
}
map> db.books.insertOne({'name':'Jerfrey','pages':'250'})
{
 acknowledged: true,
 insertedId: ObjectId('670a1665d11af492e486b021')
}
map> db.books.find().pretty()
[
  {
    _id: ObjectId('670a1607d11af492e486b01d'),
```

```
      name: 'Harry Potter',
      pages: '100'
    },
    {
      _id: ObjectId('670a1619d11af492e486b01e'),
      name: 'Percy',
      pages: '200'
    },
    {
      _id: ObjectId('670a1630d11af492e486b01f'),
      name: 'X-men',
      pages: '300'
    },
    {
      _id: ObjectId('670a164dd11af492e486b020'),
      name: 'Catwoman',
      pages: '350'
    },
    {
      _id: ObjectId('670a1665d11af492e486b021'),
      name: 'Jerfrey',
      pages: '250'
    }
]
map> var map = function()
... {
... var category;
... if(this.pages>=250)
... category='Big Books';
... else category='Small Books';
... emit(category,{name:this.name});
... };

map> var reduce = function(key, values)
... {
... var sum =0;
... values.forEach(function(doc)
... {
... sum = sum+1;
... });
... return{books:sum};};

map> var count = db.books.mapReduce(map,reduce, {out:"book_results"});
DeprecationWarning: Collection.mapReduce() is deprecated. Use an aggregation instead.
See https://docs.mongodb.com/manual/core/map-reduce for details.

map> db[count.result].find()
[
  { _id: 'Big Books', value: { books: 3 } },
  { _id: 'Small Books', value: { books: 2 } }]
```