Problem Statemnet: Implement Map reduce operation with suitable example using MongoDB.

test> use GG;

switched to db GG

GG> db.createCollection('Journel');

{ ok: 1 }

GG> db.Journel.insertOne({'book_id':1,'book_name':'JavacdOOP','amt':500,'status':'A vailable'});

{

  acknowledged: true,

  insertedId: ObjectId("6714eec52b006f1228aeadfa")

}

GG> db.Journel.insertOne({'book_id':1,'book_name':'JavaOOP','amt':400,'status':'Not Available'});

{

  acknowledged: true,

  insertedId: ObjectId("6714eee52b006f1228aeadfb")

}

GG> db.Journel.insertOne({'book_id':1,'book_name':'Java','amt':300,'s tatus':'Not Available'});

{

  acknowledged: true,

  insertedId: ObjectId("6714ef032b006f1228aeadfc")

}

GG> db.Journel.insert({'book_id':2,'book_name':'Java','amt':300,'s tatus':'Available'});

{

  acknowledged: true,

  insertedIds: { '0': ObjectId("6714ef1a2b006f1228aeadfd") }

}

```
GG> db.Journel.insert({'book_id':2,'book_name':'OPP','amt':200,'st atus':'Available'});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6714ef3a2b006f1228aeadfe") }
}
GG> db.Journal.insert({'book_id':2,'book_name':'C+','amt':200,'status':'Available'}
... );
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6714ef502b006f1228aeadff") }
}
GG> db.Journal.insert({'book_id':3,'book_name':'C+','amt':150,'status':'Available'}
... );
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6714ef6d2b006f1228aeae00") }
}
GG> db.Journel.insert({'book_id':3,'book_name':'C+','amt':200,'status':'Not Available'});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6714f0132b006f1228aeae01") }
}
GG> db.Journel.insert({'book_id':3,'book_name':'C+','amt':200,'status':'Not Available'});
{
  acknowledged: true,
  insertedIds: { '0': ObjectId("6714f04b2b006f1228aeae02") }
}
GG> db.Journel.insert({'book_id':4,'book_name':'OPP C+','amt':300,'status':'Not Available'});
{
```

```
  acknowledged: true,

  insertedIds: { '0': ObjectId("6714f06f2b006f1228aeae03") }

}

GG> db.Journel.insert({'book_id':5,'book_name':'OPP C+','amt':400,'status':'Available'});

{

  acknowledged: true,

  insertedIds: { '0': ObjectId("6714f08c2b006f1228aeae04") }

}

GG> db.Journal.insert({'book_id':5,'book_name':'C+','amt':400,'status':'Available'});

{

  acknowledged: true,

  insertedIds: { '0': ObjectId("6714f0b22b006f1228aeae05") }

}

GG> db.Journel.insert({'book_id':5,'book_name':'C++Java','amt':400,'status':'Not
Available'});

{

  acknowledged: true,

  insertedIds: { '0': ObjectId("6714f0d62b006f1228aeae06") }

}
```

//MapReduce Function:

Finding the total amounts for each unique book_id.

```
GG> var mapfunction=function(){ emit(this.book_id,this.amt)};

GG> var reducefunction=function(key,value){return Array.sum(value);};

GG> db.Journel.mapReduce(mapfunction,reducefunction,{'out':'new'});

{ result: 'new', ok: 1 }

GG> db.new.find().pretty();

[

  { _id: 1, value: 1200 },

  { _id: 3, value: 400 },
```

{ _id: 2, value: 500 },

  { _id: 5, value: 800 },

  { _id: 4, value: 300 }

]

// Finding the total counts for each unique field found in the documents of the Journel collection.

GG> var m1=function(){for(var key in this){emit(key,{"count":1});}};

GG> var r1=function(key,emits){total=0;for(var i in emits){total+=emits[i].counforreturn{"count":total};};

GG> db.Journel.mapReduce(m1,r1,{'out':'rescol'})

{ result: 'rescol', ok: 1 }

GG> db.rescol.find()

[

  { _id: 'book_id', value: { count: 10 } },

  { _id: 'book_name', value: { count: 10 } },

  { _id: '_id', value: { count: 10 } },

  { _id: 'amt', value: { count: 10 } },

  { _id: 'status', value: { count: 7 } },

  { _id: 'st atus', value: { count: 1 } },

  { _id: 's tatus', value: { count: 2 } }

]

// a count of how many times the status field appears in the Journel collection.

GG> var m1=function(){emit(this.status,1);};

GG> var r1=function(key,values){return Array.sum(values);}

GG> db.Journel.mapReduce(m1,r1,{'out':'resColllection'})

{ result: 'resColllection', ok: 1 }

GG> db.resColllection.find()

[

```
  { _id: null, value: 3 },

  { _id: 'Not Available', value: 5 },

  { _id: 'A vailable', value: 1 },

  { _id: 'Available', value: 1 }

]

GG>
```