Name:Shalini samanta

QUIZ I

Date: 28-Feb-2020

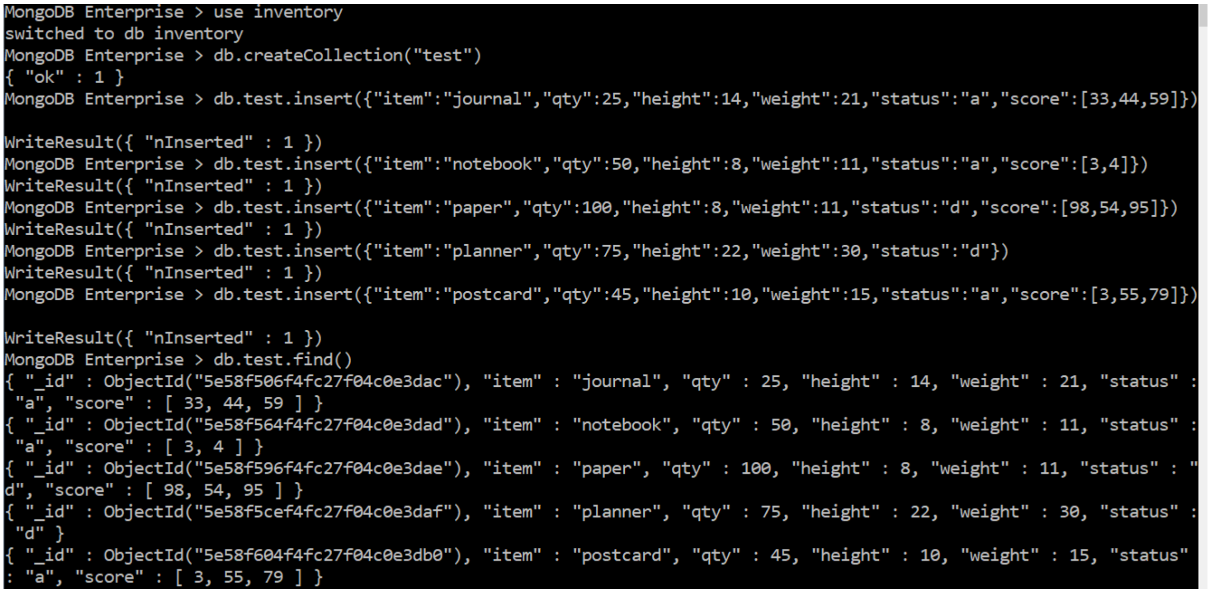
REG NO:19MCB0022

NoSOL Databases

Course Code-CSE6006 **Time =20 Minutes**

Consider the following documents in the collection test as given below

```
{item:"journal",qty:25,height:14,width:21,status:"A",score:[33,44,59]}
{item:"notebook",qty:50,height:8,width:11,status:"A",score:[3,4]}
{item:"paper",qty:100,height:8,width:11,status:"D",score:[98,54,95]}
{item:"planner",qty:75,height:22,width:30,status:"D"}
{item:"postcard",qty:45,height:10,width:15,status:"A",score:[3,55,79]}
```



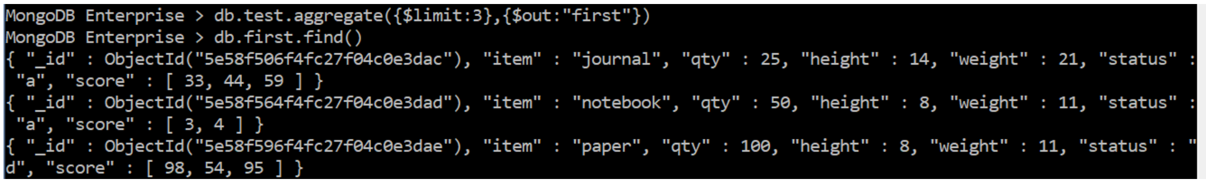
```
MongoDB Enterprise > use inventory
switched to db inventory
MongoDB Enterprise > db.createCollection("test")
{ "ok" : 1 }
MongoDB Enterprise > db.test.insert({"item":"journal","qty":25,"height":14,"weight":21,"status":"a","score":[33,44,59]})

WriteResult({ "nInserted" : 1 })
MongoDB Enterprise > db.test.insert({"item":"notebook","qty":50,"height":8,"weight":11,"status":"a","score":[3,4]})
WriteResult({ "nInserted" : 1 })
MongoDB Enterprise > db.test.insert({"item":"paper","qty":100,"height":8,"weight":11,"status":"d","score":[98,54,95]})
WriteResult({ "nInserted" : 1 })
MongoDB Enterprise > db.test.insert({"item":"planner","qty":75,"height":22,"weight":30,"status":"d"})
WriteResult({ "nInserted" : 1 })
MongoDB Enterprise > db.test.insert({"item":"postcard","qty":45,"height":10,"weight":15,"status":"a","score":[3,55,79]})

WriteResult({ "nInserted" : 1 })
MongoDB Enterprise > db.test.find()
{ "_id" : ObjectId("5e58f506f4fc27f04c0e3dac"), "item" : "journal", "qty" : 25, "height" : 14, "weight" : 21, "status" :
 "a", "score" : [ 33, 44, 59 ] }
{ "_id" : ObjectId("5e58f564f4fc27f04c0e3dad"), "item" : "notebook", "qty" : 50, "height" : 8, "weight" : 11, "status" :
 "a", "score" : [ 3, 4 ] }
{ "_id" : ObjectId("5e58f596f4fc27f04c0e3dae"), "item" : "paper", "qty" : 100, "height" : 8, "weight" : 11, "status" : "
d", "score" : [ 98, 54, 95 ] }
{ "_id" : ObjectId("5e58f5cef4fc27f04c0e3daf"), "item" : "planner", "qty" : 75, "height" : 22, "weight" : 30, "status" :
 "d" }
{ "_id" : ObjectId("5e58f604f4fc27f04c0e3db0"), "item" : "postcard", "qty" : 45, "height" : 10, "weight" : 15, "status"
: "a", "score" : [ 3, 55, 79 ] }
```

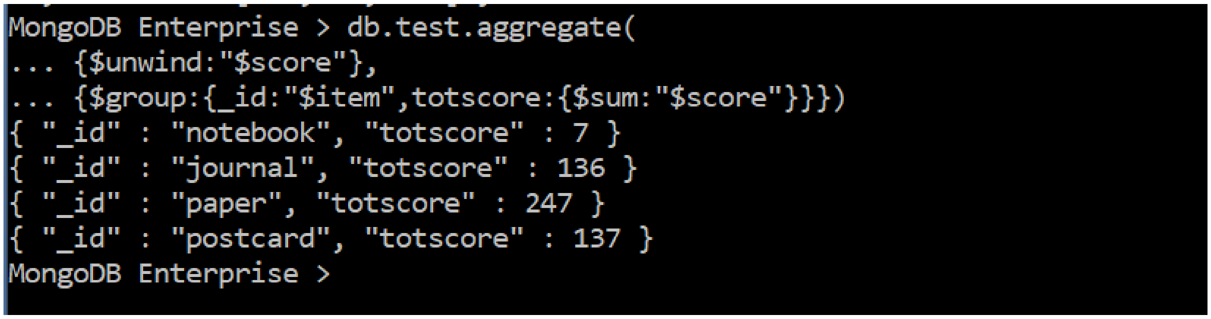
Write Mongo DB Queries for the following

1. Write the first three documents from the collection to a new collection [1 mark]



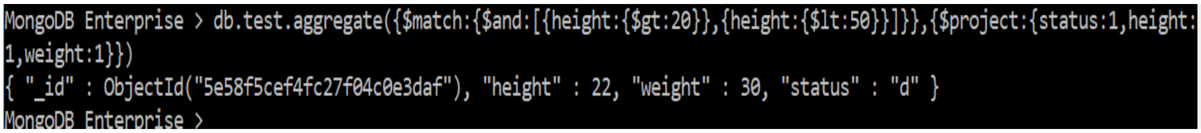
```
MongoDB Enterprise > db.test.aggregate({$limit:3},{$out:"first"})
MongoDB Enterprise > db.first.find()
{ "_id" : ObjectId("5e58f506f4fc27f04c0e3dac"), "item" : "journal", "qty" : 25, "height" : 14, "weight" : 21, "status" :
 "a", "score" : [ 33, 44, 59 ] }
{ "_id" : ObjectId("5e58f564f4fc27f04c0e3dad"), "item" : "notebook", "qty" : 50, "height" : 8, "weight" : 11, "status" :
 "a", "score" : [ 3, 4 ] }
{ "_id" : ObjectId("5e58f596f4fc27f04c0e3dae"), "item" : "paper", "qty" : 100, "height" : 8, "weight" : 11, "status" : "
d", "score" : [ 98, 54, 95 ] }
```

2. Display the total scores for each item [2 marks]



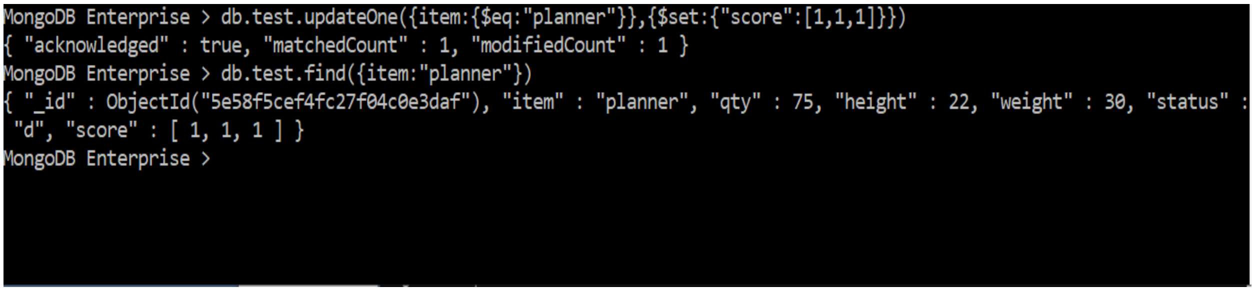
```
MongoDB Enterprise > db.test.aggregate(
... {$unwind:"$score"},
... {$group:{_id:"$item",totscore:{$sum:"$score"}}})
{ "_id" : "notebook", "totscore" : 7 }
{ "_id" : "journal", "totscore" : 136 }
{ "_id" : "paper", "totscore" : 247 }
{ "_id" : "postcard", "totscore" : 137 }
MongoDB Enterprise >
```

3. Write query to display only size and status of items whose height is greater than 20 and less than 50 from the collection[1 mark]



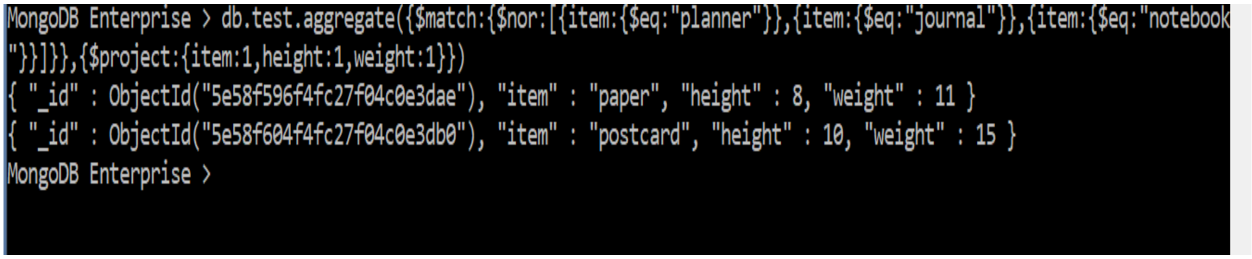
```
MongoDB Enterprise > db.test.aggregate({$match:{$and:[{height:{$gt:20}},{height:{$lt:50}}]}},{$project:{status:1,height:
1,weight:1}})
{ "_id" : ObjectId("5e58f5cef4fc27f04c0e3daf"), "height" : 22, "weight" : 30, "status" : "d" }
MongoDB Enterprise >
```

4.Set score as score:[1,1,1] only for all item:”planner” that doesn’t have a score field [1 Mark]



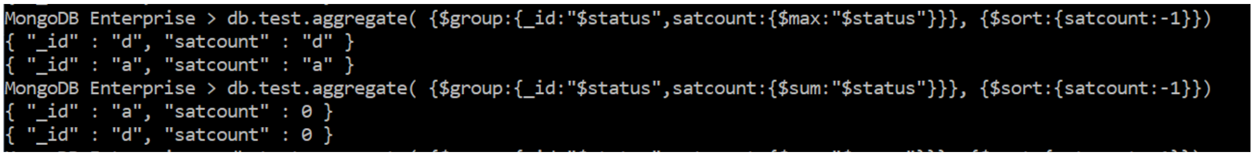
```
MongoDB Enterprise > db.test.updateOne({item:{$eq:"planner"}},{$set:{"score":[1,1,1]}})
{ "acknowledged" : true, "matchedCount" : 1, "modifiedCount" : 1 }
MongoDB Enterprise > db.test.find({item:"planner"})
{ "_id" : ObjectId("5e58f5cef4fc27f04c0e3daf"), "item" : "planner", "qty" : 75, "height" : 22, "weight" : 30, "status" :
 "d", "score" : [ 1, 1, 1 ] }
MongoDB Enterprise >
```

5. Display item,height and weight of all items that are not planner,journal or notebook[1 marks]



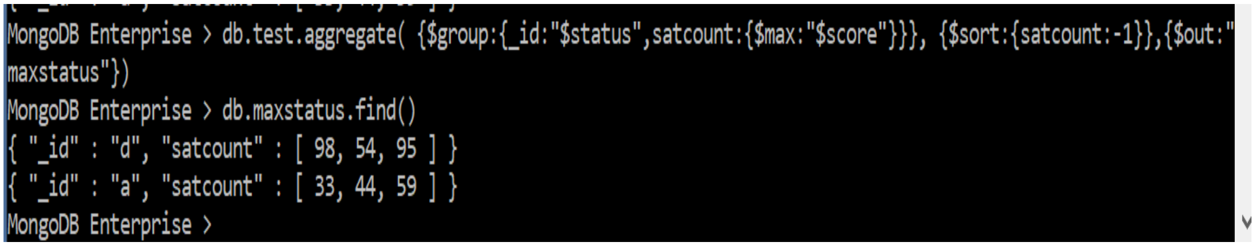
```
MongoDB Enterprise > db.test.aggregate({$match:{$nor:[{item:{$eq:"planner"}},{item:{$eq:"journal"}},{item:{$eq:"notebook
"}}]}},{$project:{item:1,height:1,weight:1}})
{ "_id" : ObjectId("5e58f596f4fc27f04c0e3dae"), "item" : "paper", "height" : 8, "weight" : 11 }
{ "_id" : ObjectId("5e58f604f4fc27f04c0e3db0"), "item" : "postcard", "height" : 10, "weight" : 15 }
MongoDB Enterprise >
```

6. Write the total count of items based on each status in the descending order to a collection by name statuscount [2 marks]



```
MongoDB Enterprise > db.test.aggregate( {$group:{_id:"$status",satcount:{$max:"$status"}}}, {$sort:{satcount:-1}})
{ "_id" : "d", "satcount" : "d" }
{ "_id" : "a", "satcount" : "a" }
MongoDB Enterprise > db.test.aggregate( {$group:{_id:"$status",satcount:{$sum:"$status"}}}, {$sort:{satcount:-1}})
{ "_id" : "a", "satcount" : 0 }
{ "_id" : "d", "satcount" : 0 }
```

8. Write the maximum score for each status to collection by name maxstatus. [2 marks].



```
MongoDB Enterprise > db.test.aggregate( {$group:{_id:"$status",satcount:{$max:"$score"}}}, {$sort:{satcount:-1}},{$out:"
maxstatus"})
MongoDB Enterprise > db.maxstatus.find()
{ "_id" : "d", "satcount" : [ 98, 54, 95 ] }
{ "_id" : "a", "satcount" : [ 33, 44, 59 ] }
MongoDB Enterprise >
```