

Project 2: MongoDB

Task 1:

1)

```
> use DATABASE_NAME
switched to db DATABASE_NAME
> use Database_Test
switched to db Database_Test
```

2)

```
> use DATABASE_NAME
switched to db DATABASE_NAME
> db.dropDatabase()
{"ok" : 1 }
```

3)

```
use Database_Test
switched to db Database_Test
db.createCollection(size : 4, autoIndexID : true, max: 1111)
2018-11-15T17:45:04.225-0700 E QUERY [js] SyntaxError: missing ) after argument list @(shell):1:25
db.createCollection("randomTest", {size : 4, autoIndexID : true, max: 1111})

{"ok" : 0,
  "errmsg" : "The field 'autoIndexID' is not a valid collection option. Options: { size: 4.0, autoIndexID: true,
max: 1111.0 }",
  "code" : 72,
  "codeName" : "InvalidOptions"

db.createCollection("randomTest", {size : 4, max: 1111})
{"ok" : 1 }
```

(Might as well show me messing around and messing up as well)

4)

```
> db.randomTest.drop()
true
```

```

db.createCollection("rando")
"ok" : 1 }
db.rando.insert({_id: ObjectId(66), title: 'aaaaaa', description: 'they let you put eggnog instead of milk at Starbucks'})
018-11-15T18:06:13.580-0700 E QUERY    [js] ReferenceError: ObjectId is not defined :
(shell):1:19
db.rando.insert({_id: ObjectId(66), title: 'aaaaaa', description: 'they let you put eggnog instead of milk at Starbucks'})
018-11-15T18:06:29.407-0700 E QUERY    [js] Error: invalid object id: length :
(shell):1:24
db.rando.insert({title: 'aaaaaa', description: 'they let you put eggnog instead of milk at Starbucks'})
writeResult({ "nInserted" : 1 })
5)
(I hope you are enjoying my flubs)

```

```

> db.rando.find().pretty()
{
  "_id" : ObjectId("5bee182f6525f90fa231099e"),
  "title" : "aaaaaa",
  "description" : "they let you put eggnog instead of milk at Starbucks"
}
6)

```

```

> db.rando.update({'description': 'they let you put eggnog instead of milk at Starbucks'}, {'description': 'seems like you dont care, so I will go away ;'})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
7)

```

```

> db.rando.remove({'_id' : ObjectId("5bee182f6525f90fa231099e")})
WriteResult({ "nRemoved" : 1 })
> db.rando.find().pretty()
{}
8)

```

(Added a find to show it is gone)

Task 2 (I know I don't need to, but I will for my own reference):

1) Source for how: <https://www.quora.com/How-can-I-get-the-lowest-values-in-a-MongoDB-collection>

```

> db.population.find({'state' : 'CO'}, {'pop': 1}).sort({'pop': 1}).limit(5)
{ "_id" : "80914", "pop" : 0 }
{ "_id" : "81633", "pop" : 10 }
{ "_id" : "81243", "pop" : 11 }
{ "_id" : "81081", "pop" : 21 }
{ "_id" : "81021", "pop" : 28 }

```

2)

```

> db.population.find({'state': 'NY', 'city': 'BROOKLYN'}, {'state': 'NY', 'city': 'BROOKLYN'}).limit(3)
{ "_id" : "11201", "city" : "BROOKLYN", "state" : "NY" }
{ "_id" : "11204", "city" : "BROOKLYN", "state" : "NY" }
{ "_id" : "11203", "city" : "BROOKLYN", "state" : "NY" }

```

3)

```
> db.population.find({'$or' : [{'state':'CA'},{'state':'NM'},{'state':'AZ'}]}).count()  
2062
```

(I used the documentation to figure out or)

4) Source for aggregation: <https://docs.mongodb.com/manual/core/aggregation-pipeline/>

```
> db.population.aggregate([{$group:{_id : '$state', sum:{$sum:'$pop'}}}])  
{ "_id" : "AK", "sum" : 544698 }  
{ "_id" : "AZ", "sum" : 3665228 }  
{ "_id" : "UT", "sum" : 1722850 }  
{ "_id" : "WY", "sum" : 453528 }  
{ "_id" : "CO", "sum" : 3293755 }  
{ "_id" : "ID", "sum" : 1006749 }  
{ "_id" : "OK", "sum" : 3145585 }  
{ "_id" : "AR", "sum" : 2350725 }  
{ "_id" : "LA", "sum" : 4217595 }  
{ "_id" : "NV", "sum" : 1201833 }  
{ "_id" : "NE", "sum" : 1578139 }  
{ "_id" : "KS", "sum" : 2475285 }  
{ "_id" : "MO", "sum" : 5110648 }  
{ "_id" : "IL", "sum" : 11427576 }  
{ "_id" : "OH", "sum" : 10846517 }  
{ "_id" : "KY", "sum" : 3675484 }  
{ "_id" : "IN", "sum" : 5544136 }  
{ "_id" : "WI", "sum" : 4891769 }  
{ "_id" : "HI", "sum" : 1108229 }  
{ "_id" : "MS", "sum" : 2573216 }
```

```
{ "_id" : "IA", "sum" : 2776420 }  
{ "_id" : "AL", "sum" : 4040587 }  
{ "_id" : "FL", "sum" : 12686644 }  
{ "_id" : "MI", "sum" : 9295297 }  
{ "_id" : "OR", "sum" : 2842321 }  
{ "_id" : "GA", "sum" : 6478216 }  
{ "_id" : "NC", "sum" : 6628637 }  
{ "_id" : "MT", "sum" : 798948 }  
{ "_id" : "ND", "sum" : 638272 }  
{ "_id" : "MD", "sum" : 4781379 }  
{ "_id" : "MN", "sum" : 4372982 }  
{ "_id" : "DC", "sum" : 606900 }  
{ "_id" : "DE", "sum" : 666168 }  
{ "_id" : "WV", "sum" : 1793146 }  
{ "_id" : "NJ", "sum" : 7730188 }  
{ "_id" : "TN", "sum" : 4876457 }  
{ "_id" : "VA", "sum" : 6181479 }  
{ "_id" : "WA", "sum" : 4866692 }  
{ "_id" : "CT", "sum" : 3287116 }  
{ "_id" : "NY", "sum" : 17990402 }
```

```
{ "_id" : "SC", "sum" : 3486703 }
{ "_id" : "PA", "sum" : 11881643 }
{ "_id" : "VT", "sum" : 562758 }
{ "_id" : "TX", "sum" : 16984601 }
{ "_id" : "SD", "sum" : 695397 }
{ "_id" : "NH", "sum" : 1109252 }
{ "_id" : "CA", "sum" : 29754890 }
{ "_id" : "NM", "sum" : 1515069 }
{ "_id" : "RI", "sum" : 1003218 }
{ "_id" : "ME", "sum" : 1226648 }
{ "_id" : "MA", "sum" : 6016425 }
>
```