

Evaluation SantéPrisonCriminalité Part 1

Exo A: Densité de population

1.

```
cd C:\Program Files\MongoDB\Server\4.0\bin  
mongoimport --jsonArray --host localhost:27017 --db europe --collection eu <  
"C:\Users\Dell\Desktop\Evaluation SantéPrisonCriminalité Part 1\Europe.json"
```

```
use europe
```

Superficie de l'europe:

```
db.eu.aggregate([{$group:{_id:,"Superficie de l'Europe":{$sum:"$Superficie(km2)"}}}]])
```

Plus petit pays:

```
db.eu.find().sort({"Superficie (km2)":1}).limit(1)
```

Plus grand pays:

```
db.eu.find().sort({"Superficie (km2)": -1}).limit(1)
```

2.

```
cd C:\Program Files\MongoDB\Server\4.0\bin>  
mongoimport --jsonArray --host localhost:27017 --db europe --collection population <  
"C:\Users\Dell\Desktop\Evaluation SantéPrisonCriminalité Part 1\demo.json"
```

```
use europe
```

Population de l'europe:

```
db.population.find({GEO:{$regex:/Union/}}).sort({"Value":-1}).limit(1)
```

Pays le moins peuplé:

```
db.population.find().sort({"Value":1}).limit(1)
```

Pays le plus peuplé:

```
db.population.remove({"GEO":/Union/})  
db.population.remove({"GEO":/Espace/})  
db.population.remove({"GEO":/Zone/})  
db.population.find().sort({"Value":-1}).limit(1)
```

Exo B: Criminalité

```
cd C:\Program Files\MongoDB\Server\4.0\bin>
```

```
mongoimport --jsonArray --host localhost:27017 --db criminalité --collection crime <  
"C:\Users\Dell\Desktop\Evaluation SantéPrisonCriminalité Part 1\Criminalité\crime.json"
```

```
use criminalité
```

1.

```
db.crime.find({TIME:2007,ICCS:/Cambriolage/}, {TIME:1, GEO:1, Value:1,  
_id:0}).sort({"Value":-1}).limit(3)
```

```
db.crime.find({TIME:2007,ICCS:/Cambriolage/}, {TIME:1, GEO:1, Value:1,  
_id:0}).sort({"Value":1}).limit(4)
```

2.

```
db.crime.find({ICCS:/Homicide/, Value:{"$exists":1}}, {TIME:1, GEO:1, Value:1,  
_id:0}).sort({"Value":1}).limit(10)
```

3.

```
db.crime.aggregate([{$group:{_id:{"GEO":"$GEO", "TIME":"$TIME"},"Le pays européen  
avec le moins de meurtres cumulés qui est de ":{$sum:"$Value"}}}, {$limit:6}])
```

4.

```
db.crime.find({Value:{"$exists":1}}, {TIME:1, GEO:1, Value:1, _id:0}).sort({"Value":-  
1}).limit(10)
```

```
db.crime.find({Value:{"$exists":1}}, {TIME:1, GEO:1, Value:1,  
_id:0}).sort({"Value":1}).limit(10)
```

Exo C: Police

```
cd C:\Program Files\MongoDB\Server\4.0\bin
```

```
mongoimport --jsonArray --host localhost:27017 --db eval --collection police <
"C:\Users\Dell\Desktop\Evaluation SantéPrisonCriminalité Part 1\police.json"
```

```
p=db.police
```

1.

```
p.aggregate([{$match:{"Value":{"$exists":1}}},{ $group: {_id:{pays:"$GEO"},
"total":{$avg:"$Value"}}},{ $sort:{"total":1}}])
```

2.

```
p.aggregate([{$match:{"Value":{"$exists":1},"TIME":2002}},{ $group: {_id:{pays:"$GEO"}
, "total":{$max:"$Value"}}},{ $sort:{"total":-1}},{ $limit:1}])
```

Exo D: Prison

```
cd C:\Program Files\MongoDB\Server\4.0\bin
```

```
mongoimport --jsonArray --host localhost:27017 --db eval --collection prison <
"C:\Users\Dell\Desktop\Evaluation SantéPrisonCriminalité Part 1\prison.json"
```

```
pr=dp.prison
```

```
pr.find().forEach( function (x) {x.Value = parseInt(x.Value); pr.save(x);})
```

```
pr.remove({ Value:NaN})
```

1.

```
pr.aggregate([{$match:{"Value":{"$exists":1}}},{ $group: {_id:{pays:"$GEO"},
"total":{$avg:"$Value"}}},{ $sort:{"total":-1}},{ $limit:1}])
```

2.

```
pr.aggregate([{$match:{"Value":{"$exists":1}}},{ $group: {_id:{pays:"$GEO"},
"total":{$avg:"$Value"}}},{ $sort:{"total":1}},{ $limit:1}])
```

3.

```
pr.aggregate([{$match:{"Value":{"$exists":1}}},{ $group: {_id:{Annee:"$TIME"},
"total":{$avg:"$Value"}}},{ $sort:{"total":-1}},{ $limit:1}])
```

Exo E: Santé

```
cd C:\Program Files\MongoDB\Server\4.0\bin
```

```
mongoimport --jsonArray --host localhost:27017 --db eval --collection sante <
"C:\Users\Dell\Desktop\Evaluation SantéPrisonCriminalité Part 1\sante.json"
```

```
sa=dp.sante
```

```
sa.find().forEach(function (x) {x.Value = parseInt(x.Value); sa.save(x);})
```

```
sa.remove({ Value:NaN})
```

1.

```
sa.aggregate([{$match:{"Value":{"$exists":1}}, UNIT:{$regex:/Euro par h/},
SPDEPS:{$regex:/Soins de/}},{$group: {_id:{pays:"$GEO", annee:"$TIME"},
"total":{$avg:"$Value"}}},{$sort:{"total":-1}},{$limit:1}])
```

2.

Requête 1:

```
var n = sa.find({TIME:2007, GEO:"Bulgarie", UNIT:{$regex:/prix constants/}}, {Value:1,
_id:0}).sort({Value:-1}).limit(1).map( function(u) { return u; } )[0].Value
```

```
var b = sa.find({TIME:2007, GEO:"Bulgarie", UNIT:{$regex:/Millions d'unites de m/}},
{Value:1, _id:0}).sort({Value:-1}).limit(1).map( function(u) { return u; } )[0].Value
```

$b/n = 1.75$ Taux de change euro Lev bulgare

$n/b = 0.57$