**PRACTICAL 5**

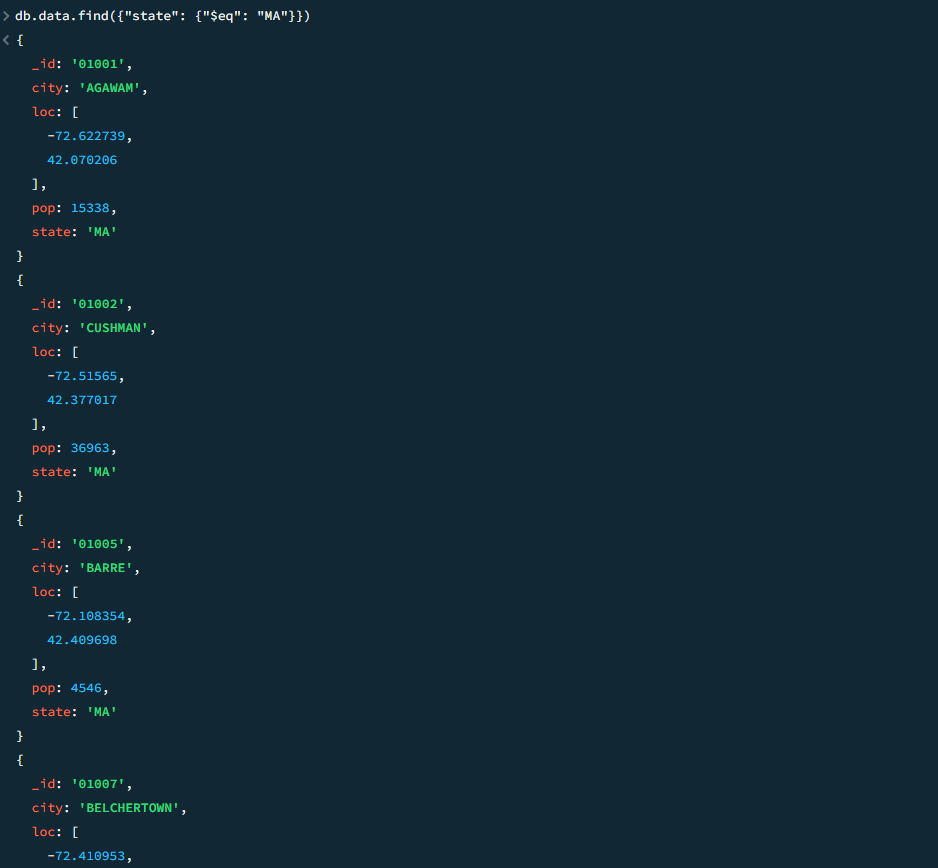
**AIM: Aggregation Using MongoBD.**

**Download and import json file in MongoDB Compass.**

**Intermediate Queries.**

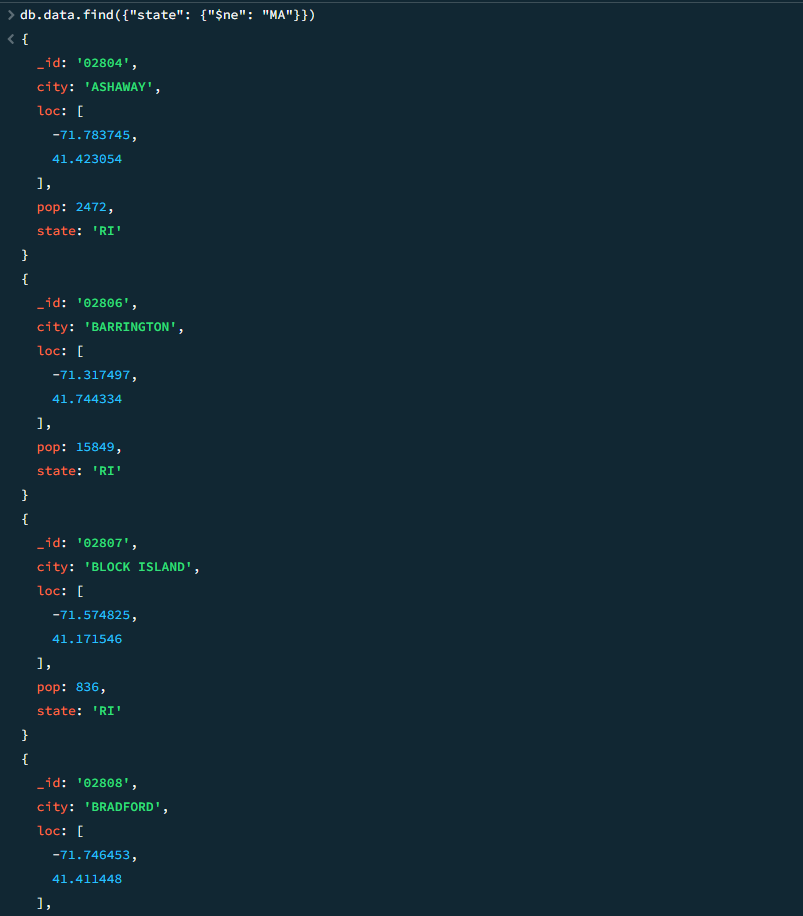
1. **Comparison Operators.**
2. **eq – Equal to**

**db.data.find({“state”: {“$eq”: “MA”}})**



1. **ne – Not equal to**

**db.data.find({“state”: {“$ne”: “MA”}})**



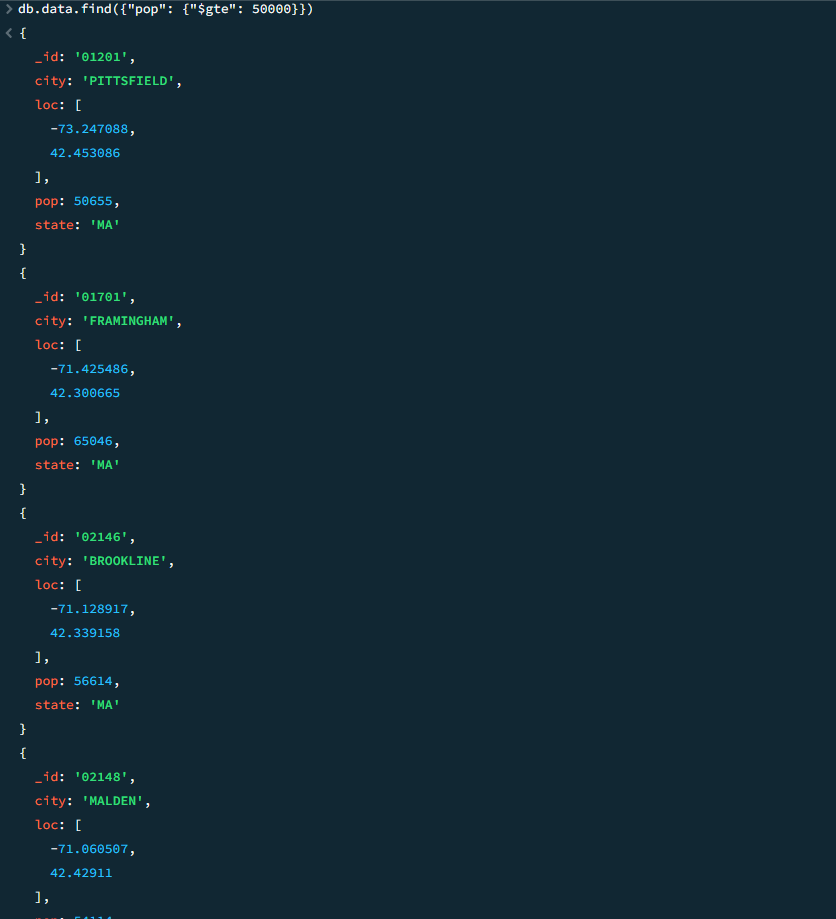
1. **gt – Greater than.**

**db.data.find({“pop”: {“$gt”: “20000”}})**



1. **gte – Greater than equal to.**

**db.data.find({“pop”: {“$gte”: “50000”}})**



1. **lt – Less than.**

**db.data.find({“loc.0”: {“$lt”: -72}})**



1. **lte – Less than equal to.**

**db.data.find({“loc.1”: {“$lte”: “42.1”}})**



1. **in**

**db.data.find({“state”: {“$in”: [“MA”, “NY”, “CA”]}})**



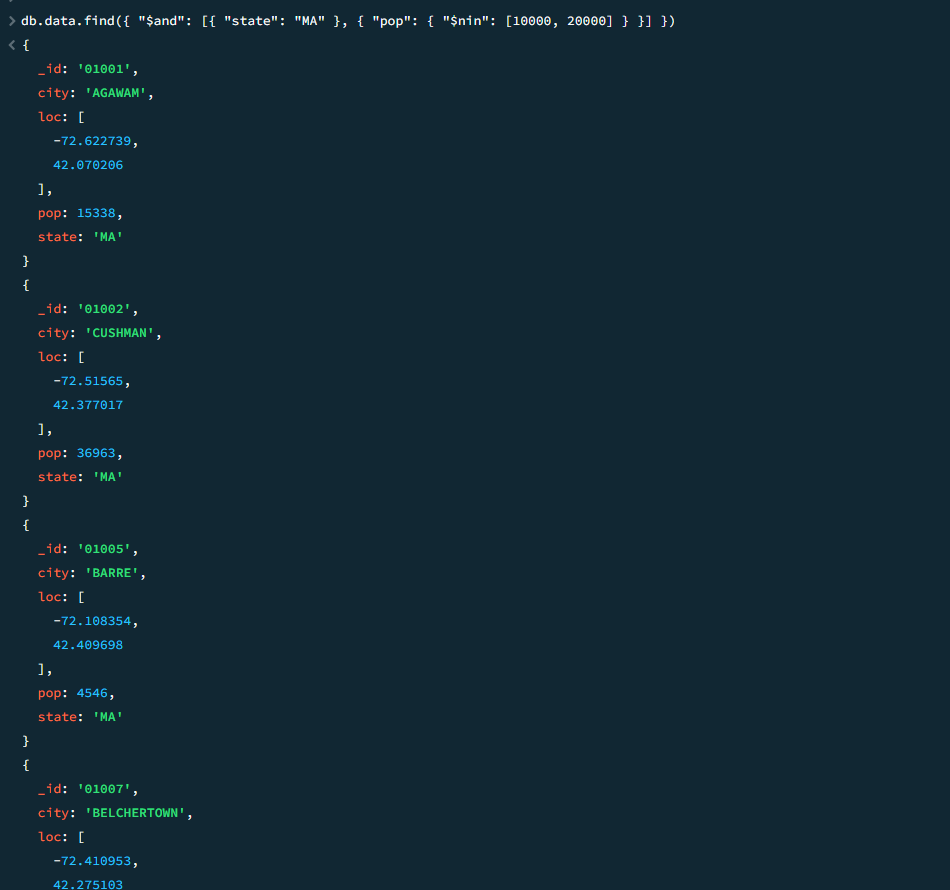
1. **nin – Not in.**

**db.data.find({“state”: {“$nin”: [“MA”, “NY”, “CA”]}})**



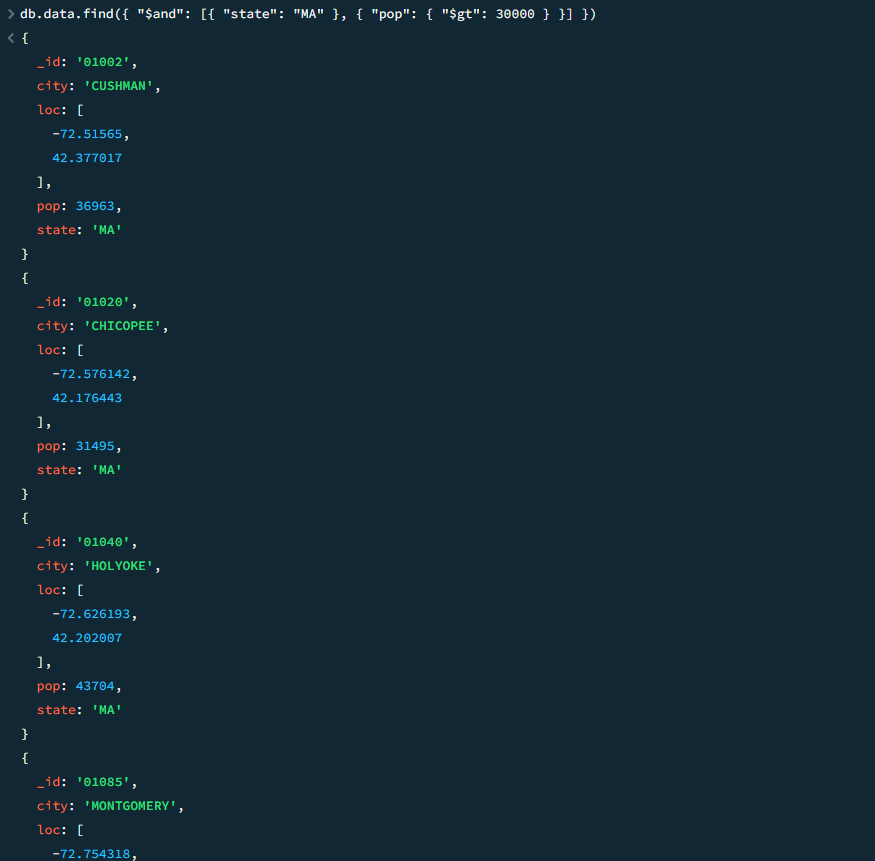
**Query with multiple Condition.**

**db.data.find({“$and”: [{“state”: “MA”}, {“pop” : { “$nin”: [10000, 20000] } }] })**



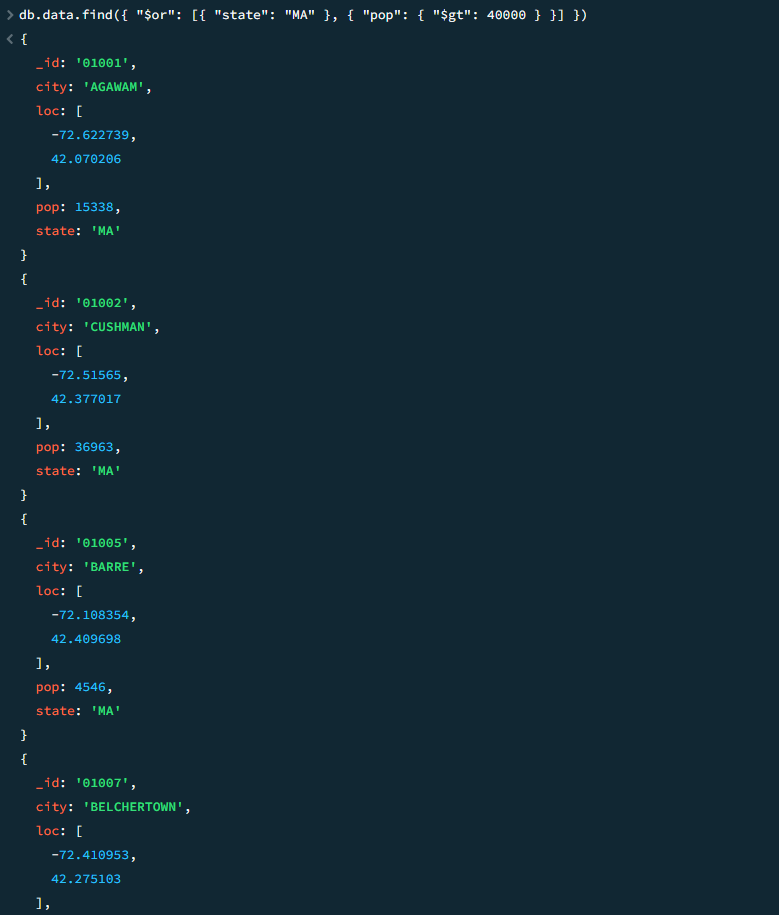
1. **Logical Operators.**
2. **and**

**db.data.find({“$and”: [{“state”: “MA”}, {“pop” : { “$gt”: 30000} }] })**



1. **or**

**db.data.find({“$or”: [{“state”: “MA”}, {“pop” : { “$gt”: 40000} }] })**



1. **ne – Not.**

**db.data.find({“state”: {“$ne”: “MA”} })**

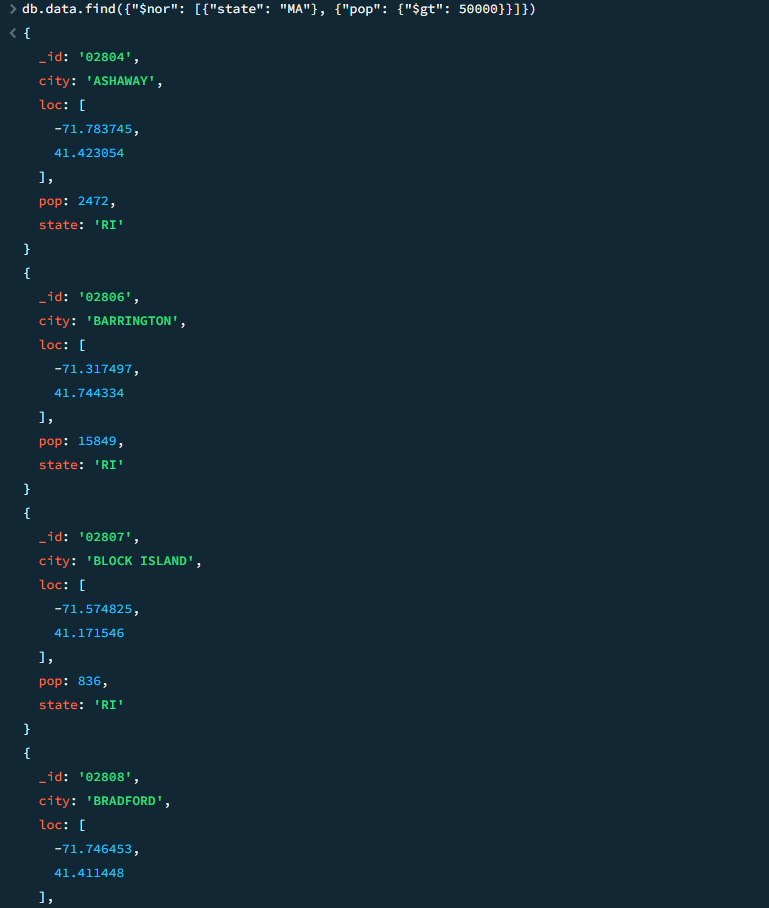
**or**

**db.data.find({“state”: {“$not”: “MA”} })**



1. **nor**

**db.data.find({“$nor”: [{“state”: “MA”}, {“pop” : { “$gt”: 50000} }] })**



1. **Element Operators.**
2. **exists**

**db.data.find({“pop”: {“$exists” : true}})**



**db.data.find({“pop”: {“$exists” : false}})**

****

1. **type**

**db.data.find({“loc”: {“$type” : “array”}})**



**db.data.find({“city”: {“$type” : “string”}})**

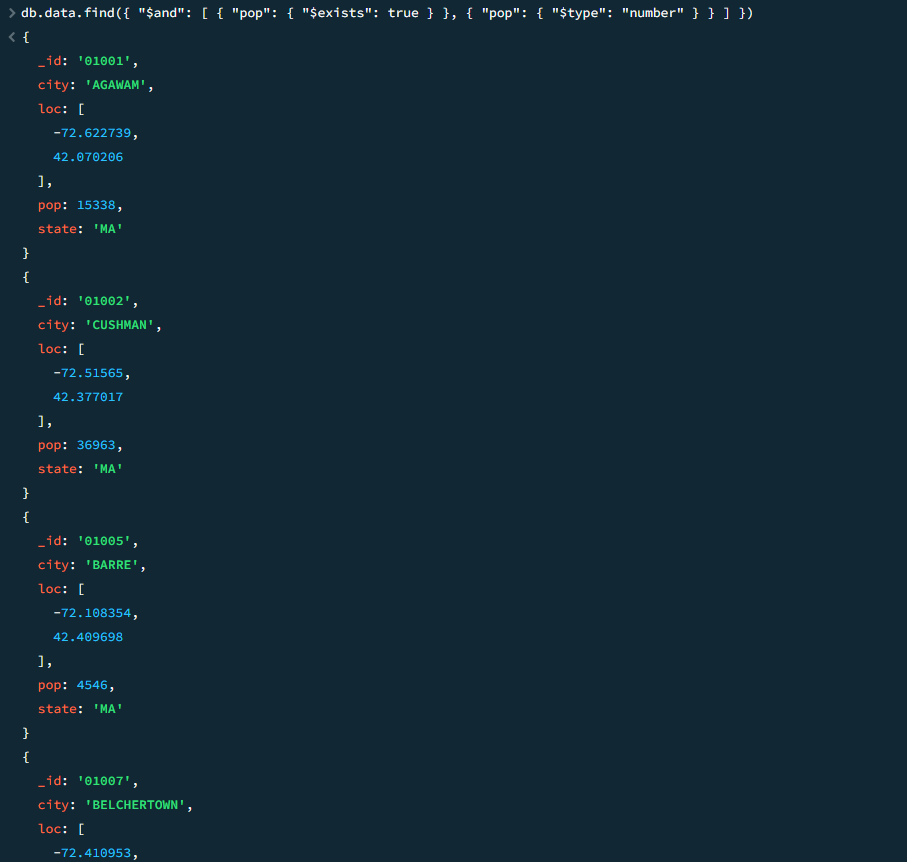


**db.data.find({“\_id”: {“$type” : “string”}})**



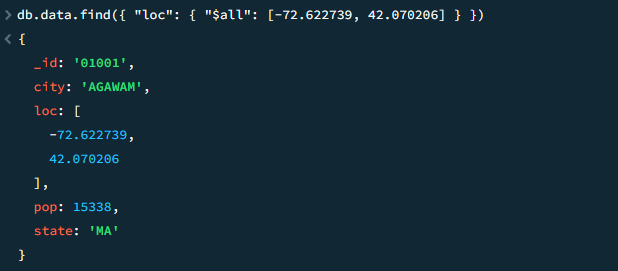
**Query with multiple Condition.**

**db.data.find({“$and”: [{“pop”: {“exists”: true }}, {“pop”: {“$type”: “number”}} ] })**



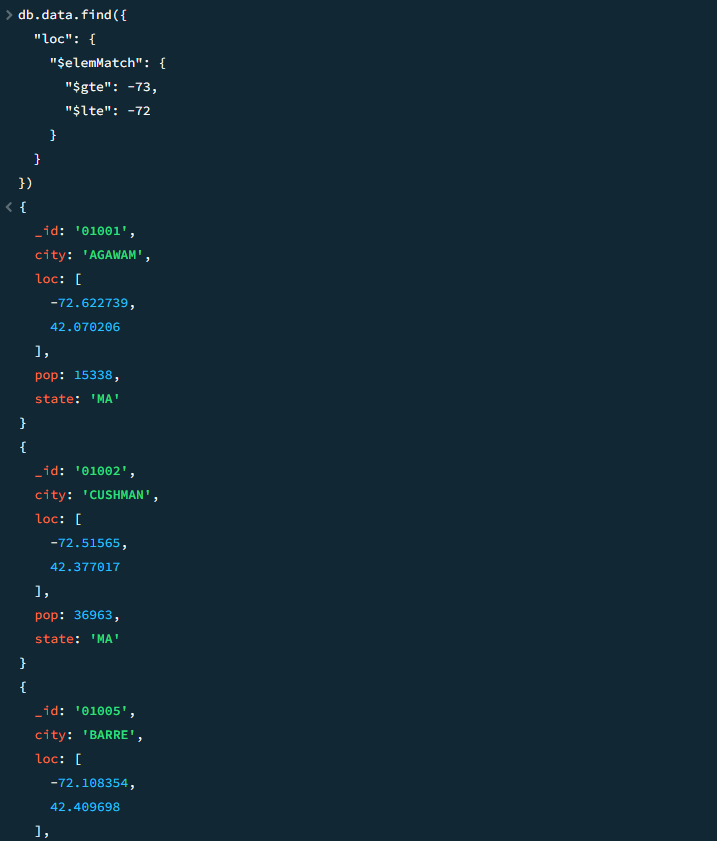
1. **Array Operators.**
2. **all**

**db.data.find({“loc”: {“$all”: [-72.622739, 42.070206] } })**



1. **eleMatch**

**db.data.find({“loc”: {“$eleMatch”: {“$gte”: -73, “$lte”: -72} }})**



1. **size**

**db.data.find({“loc”: {“$size”:2 } } )**



**Advanced Queries**

1. **group**

**db.data.aggregate([ { "$group": { "\_id": "$state", "totalPopulation": { "$sum": "$pop" }, "cityCount": { "$sum": 1 } }}])**



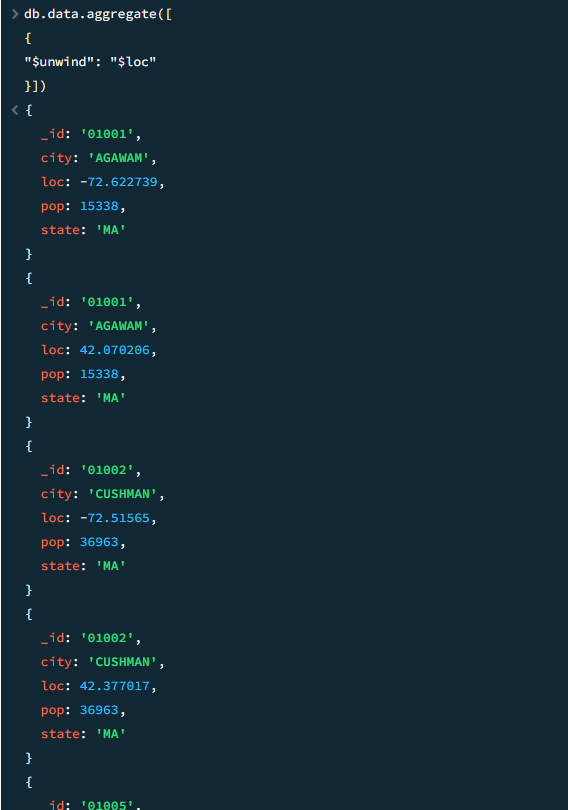
1. **sort**

**db.data.aggregate([ { "$sort": { "pop": -1 } }])**



1. **unwind**

**db.data.aggregate([ { "$unwind": "$loc" }])**



1. **match**

**db.data.aggregate([ { "$match": { "state": "MA", "pop": { "$gt": 10000 } }}])**



**Combining Operators.**

**db.data.aggregate([ { "$match": { "state": "MA" } }, { "$group": { "\_id": "$city", "totalPopulation": { "$sum": "$pop" } }}, { "$sort": { "totalPopulation": -1 } }])**

