Mithibai College
Department of Computer Science

M.Sc. (Data Science and AI)
Practical 5: Aggregation using MongoDB

Date:-31/01/2025 **Submission Date:- 07/02/2025**

Write- up:

- Comparison Operators
- Logical Operators
- Element Operators
- Array Operators

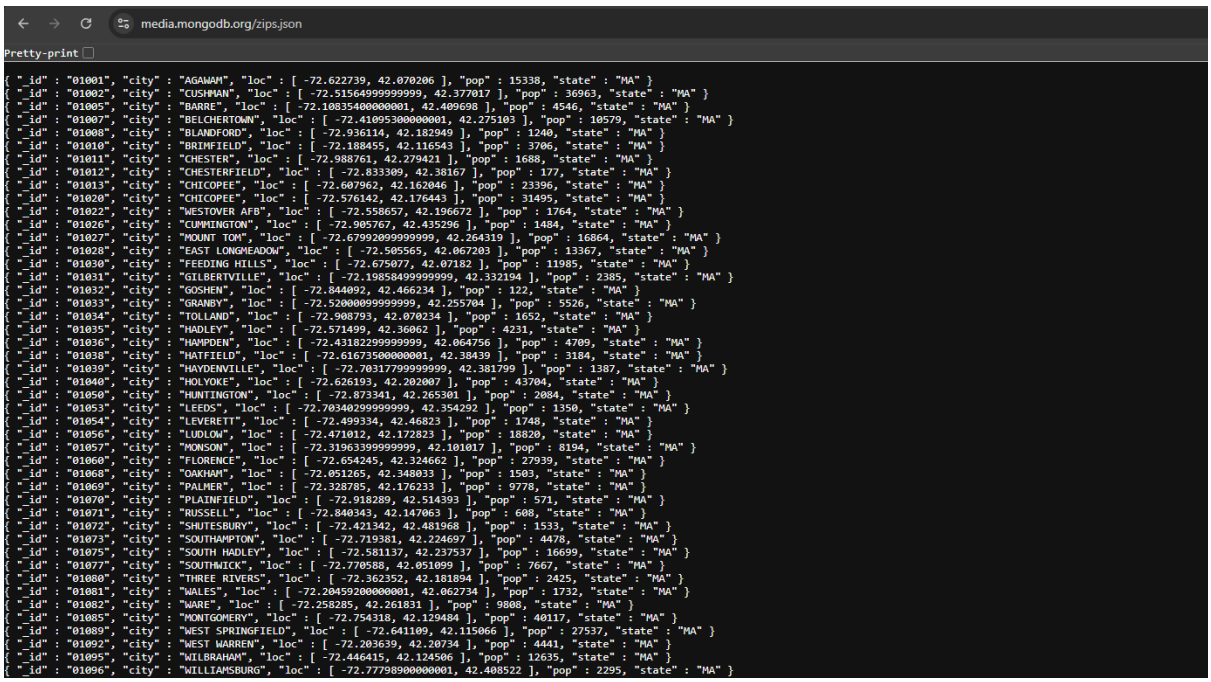
MONGOIMPORT

How to download and use mongodbimport utility
https://www.mongodb.com/try/download/database-tools
download database-tools and unzip.
Copy database tools to MongoDB bin location.
start cmd. mongoimport
2. Download sample json file from https://media.mongodb.org/zips.json

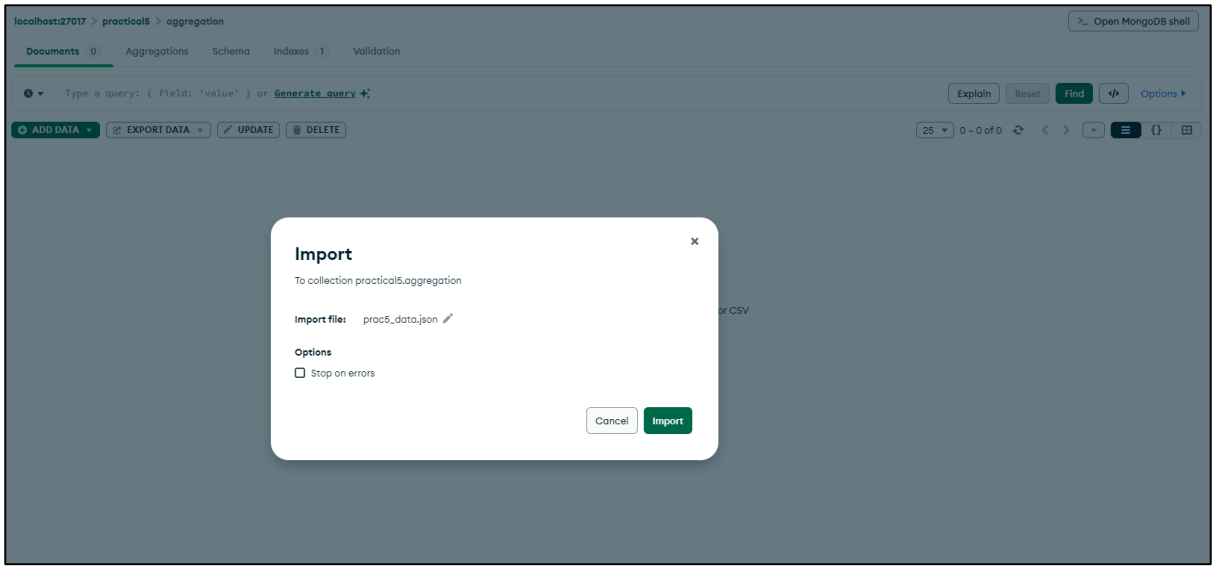
```
mongoimport --db sampledata --collection samplecollection --file C:\sample_data_from_mongodb.json
```

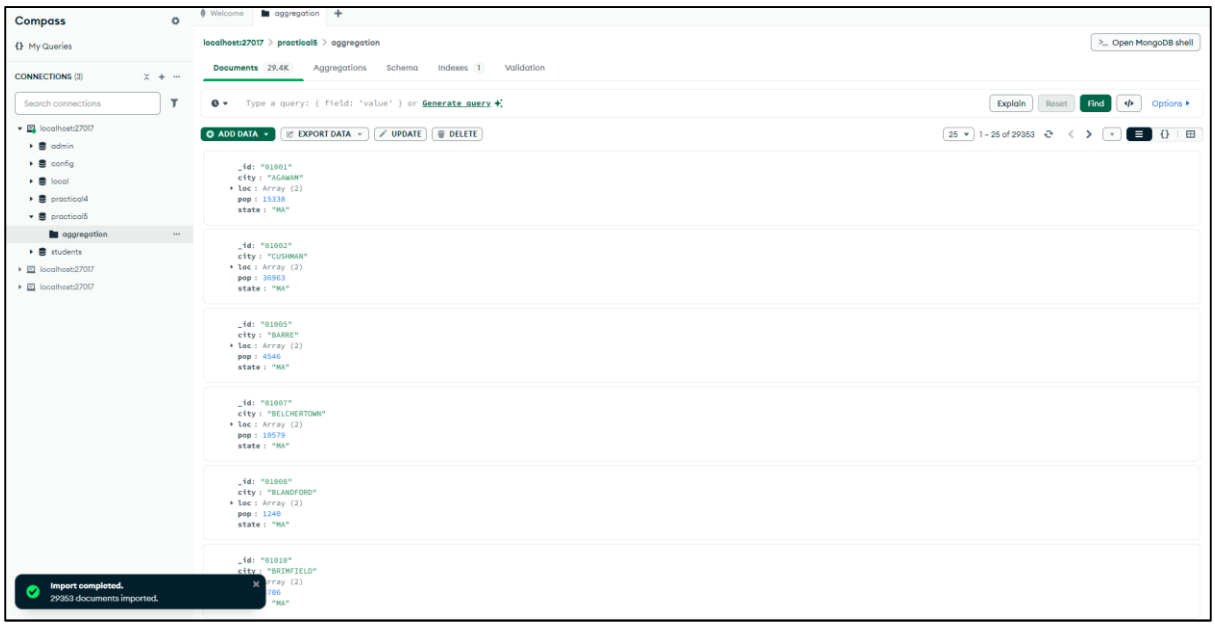
https://media.mongodb.org/zips.json

Solve the case from :
https://github.com/mattdavis0351/mongodb-labs/blob/master/exercises/02_intermediate-mongo-queries.md



{ "_id" : "01001", "city" : "AGAWAM", "loc" : [-72.622739, 42.070206], "pop" : 15338, "state" : "MA" }
{ "_id" : "01002", "city" : "CUSHMAN", "loc" : [-72.51564999999999, 42.377017], "pop" : 36963, "state" : "MA" }
{ "_id" : "01005", "city" : "BARRE", "loc" : [-72.10835400000001, 42.409698], "pop" : 4546, "state" : "MA" }
{ "_id" : "01007", "city" : "BELCHERTOWN", "loc" : [-72.41095300000001, 42.275103], "pop" : 10579, "state" : "MA" }
{ "_id" : "01008", "city" : "BLANDFORD", "loc" : [-72.936114, 42.182949], "pop" : 1240, "state" : "MA" }
{ "_id" : "01010", "city" : "BRIMFIELD", "loc" : [-72.188455, 42.116543], "pop" : 3706, "state" : "MA" }
{ "_id" : "01011", "city" : "CHESTER", "loc" : [-72.988761, 42.279421], "pop" : 1688, "state" : "MA" }
{ "_id" : "01012", "city" : "CHESTERFIELD", "loc" : [-72.833309, 42.38167], "pop" : 177, "state" : "MA" }
{ "_id" : "01013", "city" : "CHICOPEE", "loc" : [-72.607962, 42.162046], "pop" : 23396, "state" : "MA" }
{ "_id" : "01020", "city" : "CHICOPEE", "loc" : [-72.576142, 42.176443], "pop" : 31495, "state" : "MA" }
{ "_id" : "01022", "city" : "WESTOVER AFB", "loc" : [-72.558657, 42.196672], "pop" : 1764, "state" : "MA" }
{ "_id" : "01026", "city" : "CUMMINGTON", "loc" : [-72.905767, 42.435296], "pop" : 1484, "state" : "MA" }
{ "_id" : "01027", "city" : "MOUNT TOM", "loc" : [-72.67992099999999, 42.264319], "pop" : 16864, "state" : "MA" }
{ "_id" : "01028", "city" : "EAST LONGMEADOW", "loc" : [-72.505565, 42.067203], "pop" : 13367, "state" : "MA" }
{ "_id" : "01030", "city" : "FEEDING HILLS", "loc" : [-72.675077, 42.07182], "pop" : 11985, "state" : "MA" }
{ "_id" : "01031", "city" : "GILBERTVILLE", "loc" : [-72.19858499999999, 42.332194], "pop" : 2385, "state" : "MA" }
{ "_id" : "01032", "city" : "GOSHEN", "loc" : [-72.844092, 42.466234], "pop" : 122, "state" : "MA" }
{ "_id" : "01033", "city" : "GRANBY", "loc" : [-72.52000099999999, 42.255704], "pop" : 5526, "state" : "MA" }
{ "_id" : "01034", "city" : "TOLLAND", "loc" : [-72.908793, 42.070234], "pop" : 1652, "state" : "MA" }
{ "_id" : "01035", "city" : "HADLEY", "loc" : [-72.571499, 42.36062], "pop" : 4231, "state" : "MA" }
{ "_id" : "01036", "city" : "HAMPDEN", "loc" : [-72.431822999999999, 42.064756], "pop" : 4709, "state" : "MA" }
{ "_id" : "01038", "city" : "HATFIELD", "loc" : [-72.616735000000001, 42.38439], "pop" : 3184, "state" : "MA" }
{ "_id" : "01039", "city" : "HAYDENVILLE", "loc" : [-72.703177999999999, 42.381799], "pop" : 1387, "state" : "MA" }
{ "_id" : "01040", "city" : "HOLYOKE", "loc" : [-72.626193, 42.202007], "pop" : 43704, "state" : "MA" }
{ "_id" : "01050", "city" : "HUNTINGTON", "loc" : [-72.873341, 42.265301], "pop" : 2084, "state" : "MA" }
{ "_id" : "01053", "city" : "LEEDS", "loc" : [-72.703402999999999, 42.354292], "pop" : 1350, "state" : "MA" }
{ "_id" : "01054", "city" : "LEVERETT", "loc" : [-72.499334, 42.46823], "pop" : 1748, "state" : "MA" }
{ "_id" : "01056", "city" : "LUDLOW", "loc" : [-72.471012, 42.172823], "pop" : 18820, "state" : "MA" }
{ "_id" : "01057", "city" : "MONSON", "loc" : [-72.319639999999999, 42.101017], "pop" : 8194, "state" : "MA" }
{ "_id" : "01060", "city" : "FLORENCE", "loc" : [-72.654245, 42.324662], "pop" : 27939, "state" : "MA" }
{ "_id" : "01068", "city" : "OAKHAM", "loc" : [-72.051265, 42.348033], "pop" : 1503, "state" : "MA" }
{ "_id" : "01069", "city" : "PALMER", "loc" : [-72.328785, 42.176233], "pop" : 9778, "state" : "MA" }
{ "_id" : "01070", "city" : "PLAINFIELD", "loc" : [-72.918289, 42.514393], "pop" : 571, "state" : "MA" }
{ "_id" : "01071", "city" : "RUSSELL", "loc" : [-72.840343, 42.147063], "pop" : 608, "state" : "MA" }
{ "_id" : "01072", "city" : "SHUTESBURY", "loc" : [-72.421342, 42.481968], "pop" : 1533, "state" : "MA" }
{ "_id" : "01073", "city" : "SOUTHAMPTON", "loc" : [-72.719381, 42.224697], "pop" : 4478, "state" : "MA" }
{ "_id" : "01075", "city" : "SOUTH HADLEY", "loc" : [-72.581137, 42.237537], "pop" : 16699, "state" : "MA" }
{ "_id" : "01077", "city" : "SOUTHWICK", "loc" : [-72.770588, 42.051099], "pop" : 7667, "state" : "MA" }
{ "_id" : "01080", "city" : "THREE RIVERS", "loc" : [-72.362352, 42.181894], "pop" : 2425, "state" : "MA" }
{ "_id" : "01081", "city" : "WALES", "loc" : [-72.204592000000001, 42.062734], "pop" : 1732, "state" : "MA" }
{ "_id" : "01082", "city" : "WARE", "loc" : [-72.258285, 42.261831], "pop" : 9808, "state" : "MA" }
{ "_id" : "01085", "city" : "MONTGOMERY", "loc" : [-72.754318, 42.129484], "pop" : 40117, "state" : "MA" }
{ "_id" : "01089", "city" : "WEST SPRINGFIELD", "loc" : [-72.641109, 42.115066], "pop" : 27537, "state" : "MA" }
{ "_id" : "01092", "city" : "WEST WARREN", "loc" : [-72.203639, 42.20734], "pop" : 4441, "state" : "MA" }
{ "_id" : "01095", "city" : "WILBRAHAM", "loc" : [-72.446415, 42.124506], "pop" : 12635, "state" : "MA" }
{ "_id" : "01096", "city" : "WILLIAMSBURG", "loc" : [-72.777989000000001, 42.408522], "pop" : 2295, "state" : "MA" }


localhost:27017 > practical5 > aggregation
Open MongoDB shell
Documents 0
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
0 - 0 of 0
Import
To collection practical5.aggregation
Import file:
prac5_data.json
Options
Stop on errors
Cancel
Import


Compass
My Queries
CONNECTIONS (3)
Search connections
localhost:27017
admin
config
local
practical4
practical5
aggregation
students
localhost:27017
localhost:27017
Welcome
aggregation
localhost:27017 > practical5 > aggregation
Open MongoDB shell
Documents 29.4K
Aggregations
Schema
Indexes 1
Validation
Type a query: { field: 'value' } or Generate query
Explain
Reset
Find
Options
ADD DATA
EXPORT DATA
UPDATE
DELETE
25
1 - 25 of 29353
_id: "01001"
city : "AGAWAM"
loc : Array (2)
pop : 15338
state : "MA"
_id: "01002"
city : "CUSHMAN"
loc : Array (2)
pop : 36963
state : "MA"
_id: "01005"
city : "BARRE"
loc : Array (2)
pop : 4546
state : "MA"
_id: "01007"
city : "BELCHERTOWN"
loc : Array (2)
pop : 10579
state : "MA"
_id: "01008"
city : "BLANDFORD"
loc : Array (2)
pop : 1240
state : "MA"
_id: "01010"
city : "BRIMFIELD"
Import completed.
29353 documents imported.

1] Comparison Operators

Name	Description
$eq	Matches values that are equal to a specified value.
$gt	Matches values that are greater than a specified value.
$gte	Matches values that are greater than or equal to a specified value.
$in	Matches any of the values specified in an array.
$lt	Matches values that are less than a specified value.
$lte	Matches values that are less than or equal to a specified value.
$ne	Matches all values that are not equal to a specified value.
$nin	Matches none of the values specified in an array.

A] $eq

db["aggregation"].find({pop:{$eq: 1350}});

```
> db["aggregation"].find({pop:{$eq: 1350}});
< {
    _id: '01053',
    city: 'LEEDS',
    loc: [
      -72.703403,
      42.354292
    ],
    pop: 1350,
    state: 'MA'
  }
  {
    _id: '15538',
    city: 'GLENCOE',
    loc: [
      -78.827522,
      39.909046
    ],
    pop: 1350,
    state: 'PA'
  }
  {
```

B] $gt

db["aggregation"].find({pop:{$gt: 40117}});

```
> db["aggregation"].find({pop:{$gt: 40117}});
< {
    _id: '01040',
    city: 'HOLYOKE',
    loc: [
      -72.626193,
      42.202007
    ],
    pop: 43704,
    state: 'MA'
  }
  {
    _id: '01201',
    city: 'PITTSFIELD',
    loc: [
      -73.247088,
      42.453086
    ],
    pop: 50655,
    state: 'MA'
  }
  {
    _id: '01420',
    city: 'FITCHBURG',
    loc: [
      -71.803133,
      42.579563
    ],
    pop: 41194,
    state: 'MA'
  }
```

C] $lt

db["aggregation"].find({pop:{$lt: 20}});

```
> db["aggregation"].find({pop:{$lt: 20}});
< {
    _id: '01338',
    city: 'BUCKLAND',
    loc: [
      -72.764124,
      42.615174
    ],
    pop: 16,
    state: 'MA'
  }
  {
    _id: '02163',
    city: 'CAMBRIDGE',
    loc: [
      -71.141879,
      42.364005
    ],
    pop: 0,
    state: 'MA'
  }
  {
    _id: '04013',
    city: 'BUSTINS ISLAND',
    loc: [
      -70.042247,
      43.79602
    ],
    pop: 0,
    state: 'ME'
  }
```

D] $lte

db["aggregation"].find({pop:{$lte: 50000}});

```
> db["aggregation"].find({pop:{$lte: 50000}});
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
  {
    _id: '01005',
    city: 'BARRE',
    loc: [
      -72.108354,
      42.409698
    ],
    pop: 4546,
    state: 'MA'
  }
```

E] $in

db["aggregation"].find({loc:{$in: [-72.520001, 42.255704]}});

```
> db["aggregation"].find({loc:{$in: [-72.520001, 42.255704]}});
< {
    _id: '01033',
    city: 'GRANBY',
    loc: [
      -72.520001,
      42.255704
    ],
    pop: 5526,
    state: 'MA'
  }
```

F] $ne

db["aggregation"].find({state:{$ne: "MA"}});

```
> db["aggregation"].find({state:{$ne: "MA"}});
< {
    _id: '02804',
    city: 'ASHAWAY',
    loc: [
      -71.783745,
      41.423054
    ],
    pop: 2472,
    state: 'RI'
  }
  {
    _id: '02806',
    city: 'BARRINGTON',
    loc: [
      -71.317497,
      41.744334
    ],
    pop: 15849,
    state: 'RI'
  }
```

G] $nin

db["aggregation"].find({loc:{$nin: [-72.520001, 42.255704]}});

```
> db["aggregation"].find({loc:{$nin: [-72.520001, 42.255704]}});
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
```

2] Logical Operators

Name	Description
$and	Joins query clauses with a logical AND returns all documents that match the conditions of both clauses.
$or	Joins query clauses with a logical OR returns all documents that match the conditions of either clause.
$not	Inverts the effect of a query expression and returns documents that do not match the query expression.
$nor	Joins query clauses with a logical NOR returns all documents that fail to match both clauses.

A] $and

db["aggregation"].find({$and:[{state: {$eq: "MA"}}, {city: {$eq: "HADLEY"}}]});

```
> db["aggregation"].find({$and:[{state: {$eq: "MA"}}, {city: {$eq: "HADLEY"}}]});
< {
    _id: '01035',
    city: 'HADLEY',
    loc: [
      -72.571499,
      42.36062
    ],
    pop: 4231,
    state: 'MA'
  }
practical5>
```

B] $or

db["aggregation"].find({$or:[{state: {$eq: "MA"}}, {city: {$eq: "HADLEY"}}]});

```
> db["aggregation"].find({$or:[{state: {$eq: "MA"}}, {city: {$eq: "HADLEY"}}]});
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
```

```
  {
    _id: '01035',
    city: 'HADLEY',
    loc: [
      -72.571499,
      42.36062
    ],
    pop: 4231,
    state: 'MA'
  }
```

C] $not

db["aggregation"].find({$and: [{state: {$not: { $eq: "MA" } } },{city: {$not: { $eq: "HADLEY"}}}]});

```
> db["aggregation"].find({
    $and: [
      { state: { $not: { $eq: "MA" } } },
      { city: { $not: { $eq: "HADLEY" } } }
    ]
  });
< {
    _id: '02804',
    city: 'ASHAWAY',
    loc: [
      -71.783745,
      41.423054
    ],
    pop: 2472,
    state: 'RI'
  }
  {
    _id: '02806',
    city: 'BARRINGTON',
    loc: [
      -71.317497,
      41.744334
    ],
    pop: 15849,
    state: 'RI'
  }
```

D] $nor

db["aggregation"].find({$nor:[{state: {$eq: "MA"}}, {city: {$eq: "HADLEY"}}]});

```
> db["aggregation"].find({$nor:[{state: {$eq: "MA"}}, {city: {$eq: "HADLEY"}}]});
< {
    _id: '02804',
    city: 'ASHAWAY',
    loc: [
      -71.783745,
      41.423054
    ],
    pop: 2472,
    state: 'RI'
  }
  {
    _id: '02806',
    city: 'BARRINGTON',
    loc: [
      -71.317497,
      41.744334
    ],
    pop: 15849,
    state: 'RI'
  }
  {
    _id: '02807',
    city: 'BLOCK ISLAND',
    loc: [
      -71.574825,
      41.171546
    ],
    pop: 836,
    state: 'RI'
  }
```

3] Element Operators

Name	Description
$exists	Matches documents that have the specified field.
$type	Selects documents if a field is of the specified type.

A] $exists

db["aggregation"].find({state: { $exists: true }});

```
> db["aggregation"].find({state: { $exists: true }});
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
  {
    _id: '01005',
    city: 'BARRE',
    loc: [
      -72.108354,
      42.409698
    ],
    pop: 4546,
    state: 'MA'
  }
```

B] $type

db["aggregation"].find({city: { $type: "string" }});

```
> db["aggregation"].find({city: { $type: "string" }});
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
  {
    _id: '01005',
    city: 'BARRE',
    loc: [
      -72.108354,
      42.409698
    ],
    pop: 4546,
    state: 'MA'
  }
```

4] Array Operators

Name	Description
$all	Matches arrays that contain all elements specified in the query.
$elemMatch	Selects documents if element in the array field matches all the specified $elemMatch conditions.
$size	Selects documents if the array field is a specified size.

A] $all

db["aggregation"].find({loc: {$all:[-72.571499, 42.36062]}});

```
> db["aggregation"].find({loc: {$all:[-72.571499, 42.36062]}});
< {
    _id: '01035',
    city: 'HADLEY',
    loc: [
      -72.571499,
      42.36062
    ],
    pop: 4231,
    state: 'MA'
  }
```

B] $elemMatch

db["aggregation"].find({loc: {$elemMatch: { $gt: 42, $gt:50 }}});

```
> db["aggregation"].find({loc: {$elemMatch: { $gt: 42, $gt:50 }}});
< {
    _id: '99501',
    city: 'ANCHORAGE',
    loc: [
      -149.876077,
      61.211571
    ],
    pop: 14436,
    state: 'AK'
  }
```

C] $size

db["aggregation"].find({loc:{$size: 2}});

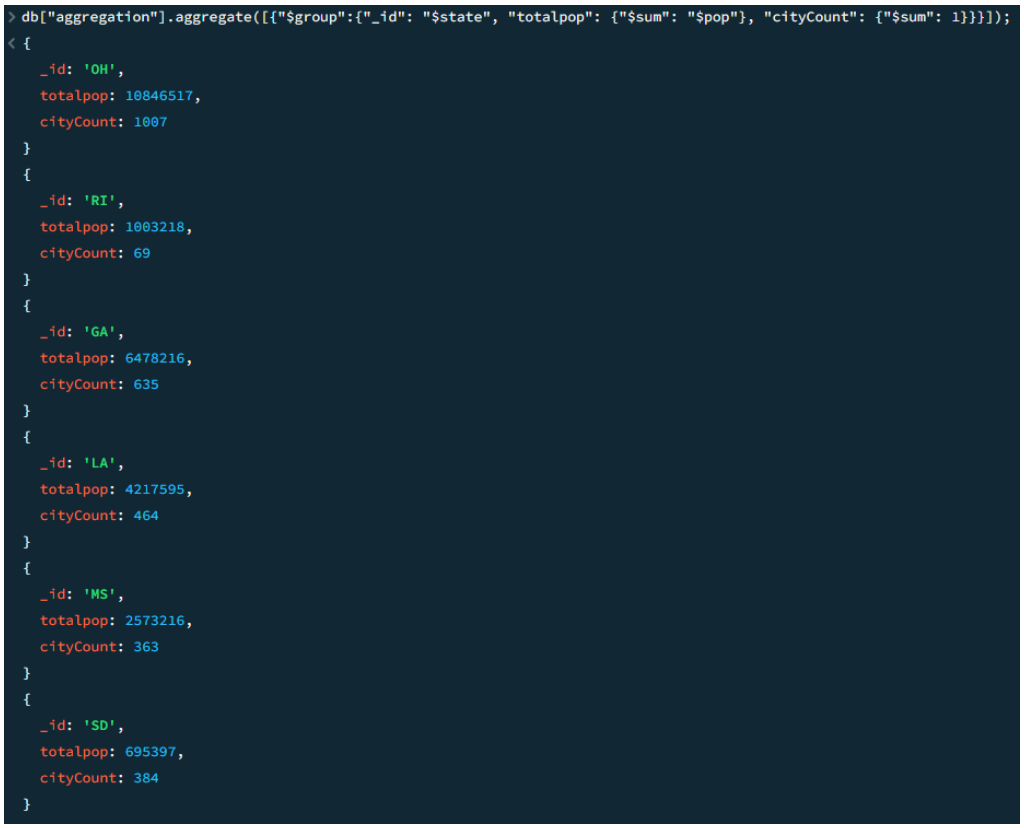
```
> db["aggregation"].find({loc:{$size: 2}});
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
  {
    _id: '01005',
    city: 'BARRE',
    loc: [
      -72.108354,
      42.409698
    ],
    pop: 4546,
    state: 'MA'
  }
```

Advanced Queries

1] The $group operator

Group cities by state and calculate the total population per state

db["aggregation"].aggregate([{"$group":{"_id": "$state", "totalpop": {"$sum": "$pop"}, "cityCount": {"$sum": 1}}}]);



2] The $match operator

Find cities in MA with a population greater than 15000

db["aggregation"].aggregate([{$match: {"state":"MA", "pop":{$gt: 15000}}}]);

```
> db["aggregation"].aggregate([{$match: {"state":"MA", "pop":{$gt: 15000}}}]);
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
```

3] The $sort operator

Sort cities by population in ascending order

db["aggregation"].aggregate([{$match: {"state":"MA", "pop":{$gt: 15000}}}]);


```
> db["aggregation"].aggregate([{$match: {"state":"MA", "pop":{$gt: 15000}}}]);
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: [
      -72.622739,
      42.070206
    ],
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: [
      -72.51565,
      42.377017
    ],
    pop: 36963,
    state: 'MA'
  }
```

4] The $unwind operator

Unwind the loc array and display the coordinate separately

db["aggregation"].aggregate([{$unwind: "$loc"}]);

```
> db["aggregation"].aggregate([{$unwind: "$loc"}]);
< {
    _id: '01001',
    city: 'AGAWAM',
    loc: -72.622739,
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01001',
    city: 'AGAWAM',
    loc: 42.070206,
    pop: 15338,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: -72.51565,
    pop: 36963,
    state: 'MA'
  }
  {
    _id: '01002',
    city: 'CUSHMAN',
    loc: 42.377017,
    pop: 36963,
    state: 'MA'
  }
```

Combining operators

Find cities in RI, group by city, and sort by total population

db["aggregation"].aggregate([{$match: {"state":"RI"}}, {$group: {"_id": "$city", "totalpop": {$sum: "$pop"}}},{$sort:{"totalpop":1}}]);

```
> db["aggregation"].aggregate([{$match: {"state":"RI"}}, {$group: {"_id": "$city", "totalpop": {$sum: "$pop"}}},{$sort:{"totalpop":1}}]);
< {
    _id: 'CLAYVILLE',
    totalpop: 45
  }
  {
    _id: 'PRUDENCE ISLAND',
    totalpop: 150
  }
  {
    _id: 'OAKLAND',
    totalpop: 462
  }
  {
    _id: 'SLATERSVILLE',
    totalpop: 639
  }
  {
    _id: 'WOOD RIVER JUNCT',
    totalpop: 684
  }
  {
    _id: 'BLOCK ISLAND',
    totalpop: 836
  }
  {
    _id: 'SLOCUM',
    totalpop: 1114
  }
```