

# Mithibai College Department of Computer Science Msc(Data Sci and AI)

---

## Practical-5: Aggregation USING Mongodb

**Date: 31/01/2025**

**Submission Date: 7/01/2025**

Write- up:

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

MONGOIMPORT

How to download and use mongodimport 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
```

Solve the case from:

[https://github.com/mattdavis0351/mongodb-labs/blob/master/exercises/02\\_intermediate-mongo-queries.md](https://github.com/mattdavis0351/mongodb-labs/blob/master/exercises/02_intermediate-mongo-queries.md)

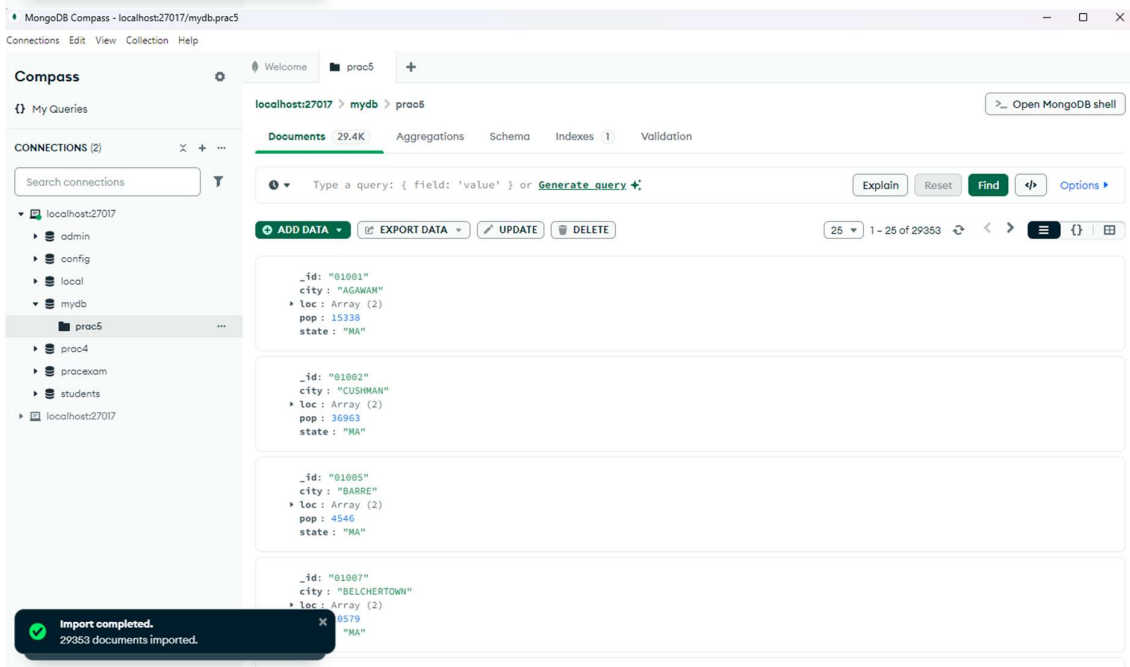
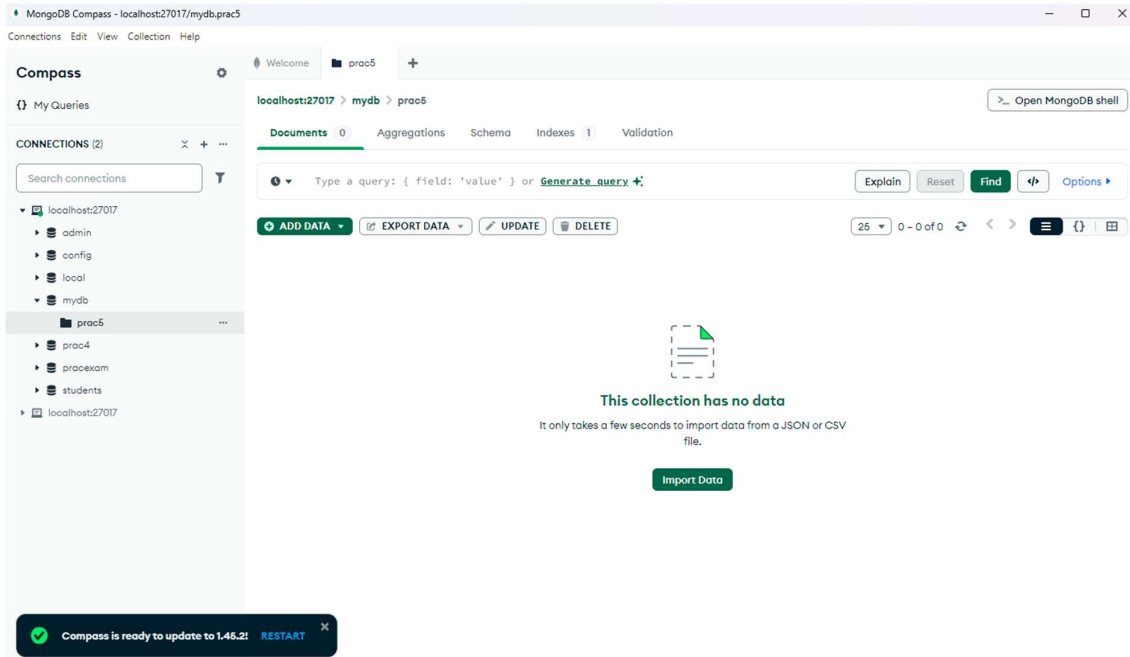
MONGOIMPORT

How to download and use mongodimport 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
```

Download the json file and import to mongodb



```
>_MONGOSH
> use mydb
< switched to db mydb
> db["prac5"].find()
< {
  _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'
}
{
```

## Intermediate Queries

### Comparison Operator:

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.

Name	Description
\$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.

eq:

```
> db.prac5.find({pop: {"$eq":5526}})
< {
  _id: '01033',
  city: 'GRANBY',
  loc: [
    -72.520001,
    42.255704
  ],
  pop: 5526,
  state: 'MA'
}
{
  _id: '07830',
  city: 'CALIFON',
  loc: [
    -74.815218,
    40.716209
  ],
  pop: 5526,
  state: 'NJ'
}
```

gt:

```
> db.prac5.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'  
}  
}
```

gte:

```
> db.prac5.find({pop: {"$gte":47687}})
< {
  _id: '01201',
  city: 'PITTSFIELD',
  loc: [
    -73.247088,
    42.453086
  ],
  pop: 50655,
  state: 'MA'
}
{
  _id: '01701',
  city: 'FRAMINGHAM',
  loc: [
    -71.425486,
    42.300665
  ],
  pop: 65046,
  state: 'MA'
}
{
  _id: '02124',
  city: 'DORCHESTER',
  loc: [
    -71.072898,
    42.287984
  ],
  pop: 48560,
  state: 'MA'
}
```

It:

```

> db.prac5.find({pop: {"$lt":5}})
< {
  _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'
}
{
  _id: '04570',
  city: 'SQUIRREL ISLAND',
  loc: [
    -69.630974,
    43.809031
  ],
  pop: 3,
  state: 'ME'
}

```

in:

```

> db.prac5.find({loc: {"$in": [81.440489, 31.979085]}})
< {
  _id: '31314',
  city: 'FORT STEWART',
  loc: [
    -81.440489,
    31.979085
  ],
  pop: 0,
  state: 'GA'
}

```

ne:

```

> db.prac5.find({city: {"$ne": "Fort Stewart"}})
< {
  _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'
}

```

**nin:**

```

> db.prac5.find({loc: {"$nin": [81.440489, 31.979085]}})
< {
  _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'
}

```



### 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.

### and:

```
> db.prac5.find({state: {"$ne": "Hadley"}, state: {"$ne": "Tolland"}})
< {
  _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'
}
```

```
> db.prac5.find({"$and": {state: {"$ne": "Hadley"}, state: {"$ne": "Tolland"}}})
```

```
✖ • MongoServerError[BadValue]: $and must be an array
```

```
> db.prac5.find({"$and": [{state: {"$ne": "Hadley"}, state: {"$ne": "Tolland"}}]})
```

```
< {
  _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'
}
```

OR:

```
> db.prac5.find({"$or": [{state: {"$ne": "Hadley"}, state: {"$ne": "Tolland"}}]})
```

```
< {
  _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'
}
{
  _id: '01006',
  city: 'MIDDLETOWN',
  loc: [
    -72.655556,
    42.055556
  ],
  pop: 15000,
  state: 'MA'
}
```

```

> db.prac5.find({"$or": [{state: {"$ne": "MA"}}, {city: {"$ne": "Tolland"}}]})
< {
  _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'
}
{

```

**not:**

```

> db.prac5.find({"$and": [{state: {"$not": {"$eq": "MA"}}}, {city: {"$not": {"$eq": "Tolland"}}}]});
< {
  _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'
}

```

**nor:**

```
> db.prac5.find({"$nor": [{state: {"$ne": "MA"}, city: {"$ne": "Tolland"}}]})
< {
  _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'
}
```

**Element Operators:**

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

**exists:**

```

> db.prac5.find({"city": {"$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'
}

> db.prac5.count({"city": {"$exists": true}});
< 29353

> db.prac5.count({"India": {"$exists": true}});
< DeprecationWarning: Collection.count() is deprecated
< 0

```

type:

```

> db.prac5.find({"city": {"$type": "string"}}).pretty();
< {
  _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'
}

> db.prac5.count({"city": {"$type": "string"}});
< 29353

```

### 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.

**all:**

```
> db.prac5.find({loc: {"$all": [-72.571499, 42.36062]}})
< {
  _id: '01035',
  city: 'HADLEY',
  loc: [
    -72.571499,
    42.36062
  ],
  pop: 4231,
  state: 'MA'
}
```

**elemMatch:**

```
> db.prac5.findOne({loc: {"$elemMatch": {$gt: 42, $gt:50}}})
< {
  _id: '99501',
  city: 'ANCHORAGE',
  loc: [
    -149.876077,
    61.211571
  ],
  pop: 14436,
  state: 'AK'
}
```

**size:**

```

> db.prac5.findOne({loc: {"$size": 2}})
< {
  _id: '01001',
  city: 'AGAWAM',
  loc: [
    -72.622739,
    42.070206
  ],
  pop: 15338,
  state: 'MA'
}
> db.prac5.find({loc: {"$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'
}

```

**Advance Queries:**

**\$group (Grouping Documents)**

**Group cities by state and calculate the total population per state:**



```
> db.prac5.aggregate([{$group: {"_id": "$state", "totalPopulation": {"$sum": "$pop"}, "cityCount": {"$sum": "1"}}}])
< [
  {
    _id: 'KS',
    totalPopulation: 2475285,
    cityCount: 715
  },
  {
    _id: 'NC',
    totalPopulation: 6628637,
    cityCount: 785
  },
  {
    _id: 'VT',
    totalPopulation: 562758,
    cityCount: 243
  },
  {
    _id: 'FL',
    totalPopulation: 12686644,
    cityCount: 804
  },
  {
    _id: 'NJ',
    totalPopulation: 7730188,
    cityCount: 540
  },
  {
    _id: 'ND',
    totalPopulation: 638272,
    cityCount: 391
  },
  {
    _id: 'AZ',
    totalPopulation: 3665228,
    cityCount: 270
  }
]
```

## \$sort (Sorting Documents)

Sort cities by population in descending order

```

> db.prac5.aggregate([{"$sort": {"pop": -1}}])
< {
  _id: '60623',
  city: 'CHICAGO',
  loc: [
    -87.7157,
    41.849015
  ],
  pop: 112047,
  state: 'IL'
}
{
  _id: '11226',
  city: 'BROOKLYN',
  loc: [
    -73.956985,
    40.646694
  ],
  pop: 111396,
  state: 'NY'
}
{
  _id: '10021',
  city: 'NEW YORK',
  loc: [
    -73.958805,
    40.768476
  ],
  pop: 106564,
  state: 'NY'
}

```

### \$unwind (Deconstructing Arrays)

Unwind the “loc” array and display each coordinate separately

```
> db.prac5.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'
}
{
  _id: '01005',
  city: 'BARRE',
  loc: -72.108354,
  pop: 4546,
  state: 'MA'
}
```

### \$match (Filtering Documents)

Find cities in Massachusetts (MA) with a population greater than 10,000

```
> db.prac5.aggregate([{"$match" : {"state":"MA", "pop": {"$gt": 10000}}]})
< {
  _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: '01007',
  city: 'BELCHERTOWN',
  loc: [
    -72.410953,
    42.275103
  ],
  pop: 10579,
  state: 'MA'
}
{
  _id: '01013',
  city: 'CHICOPEE',
  loc: [
    -72.607962,
```

## Combining Operators

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

```
> db.prac5.aggregate([{"$match" : {"state":"MA"}}, {"$group":{"_id":"$city","totalPopulation":{"$sum":"$pop"}}}, {"$sort": {"totalPopulation": -1}}])
< {
  _id: 'WORCESTER',
  totalPopulation: 169856
}
{
  _id: 'SPRINGFIELD',
  totalPopulation: 148062
}
{
  _id: 'DORCHESTER',
  totalPopulation: 126821
}
{
  _id: 'LOWELL',
  totalPopulation: 103439
}
{
  _id: 'NEW BEDFORD',
  totalPopulation: 99747
}
{
  _id: 'BROCKTON',
  totalPopulation: 92788
}
{
  _id: 'FALL RIVER',
  totalPopulation: 92426
}
{
  _id: 'BOSTON',
  totalPopulation: 91302
}
{
  _id: 'QUINCY',
  totalPopulation: 85501
}
}
```