MONGODB – Assignment -2

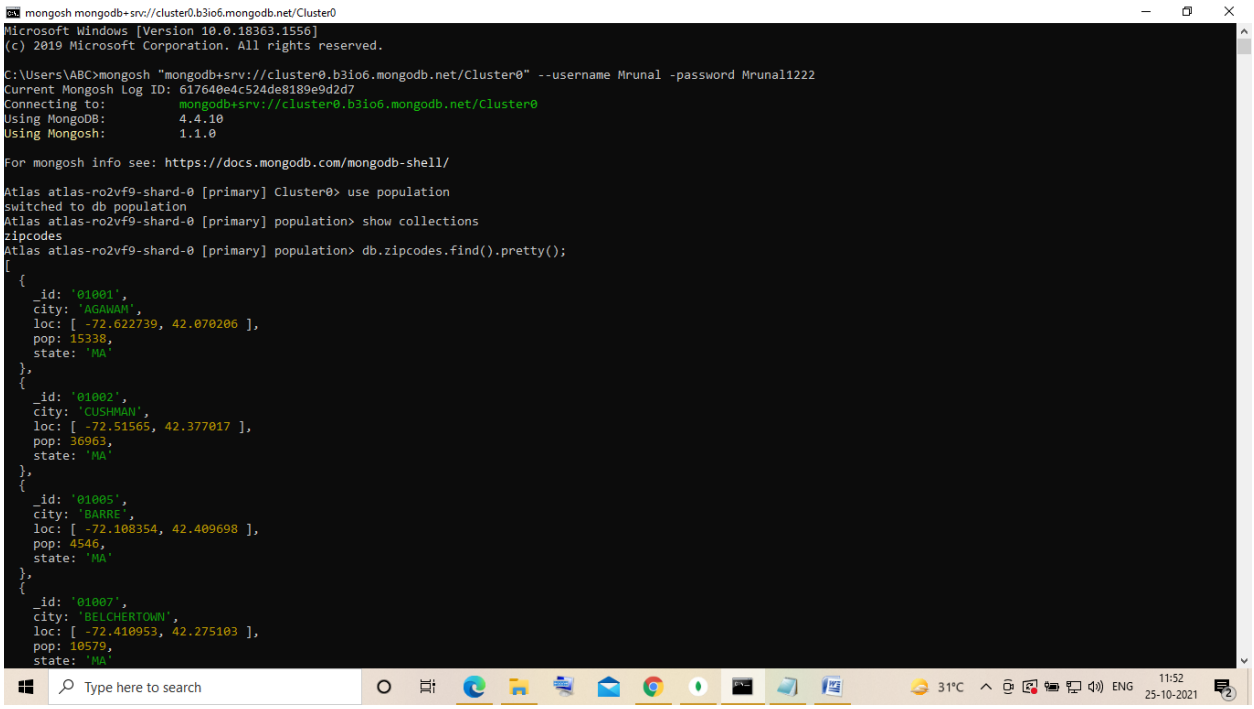
Connect with Compass app -> create a database -> enter db name(population) and collection name(zipcodes) ->create -> import data ->select JSON and browse the zip file(zips.json) -> import

C:\Users\ABC>mongosh "mongodb+srv://cluster0.b3io6.mongodb.net/Cluster0" --username Mrunal -password Mrunal1222

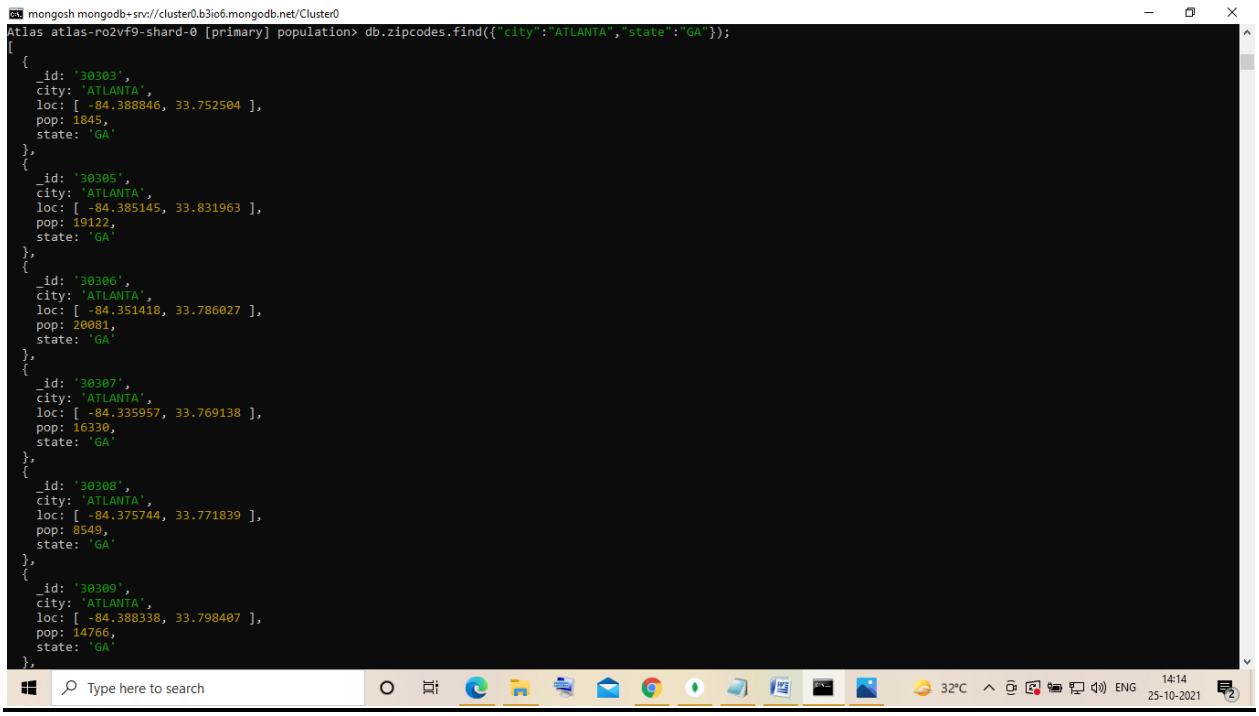
use population

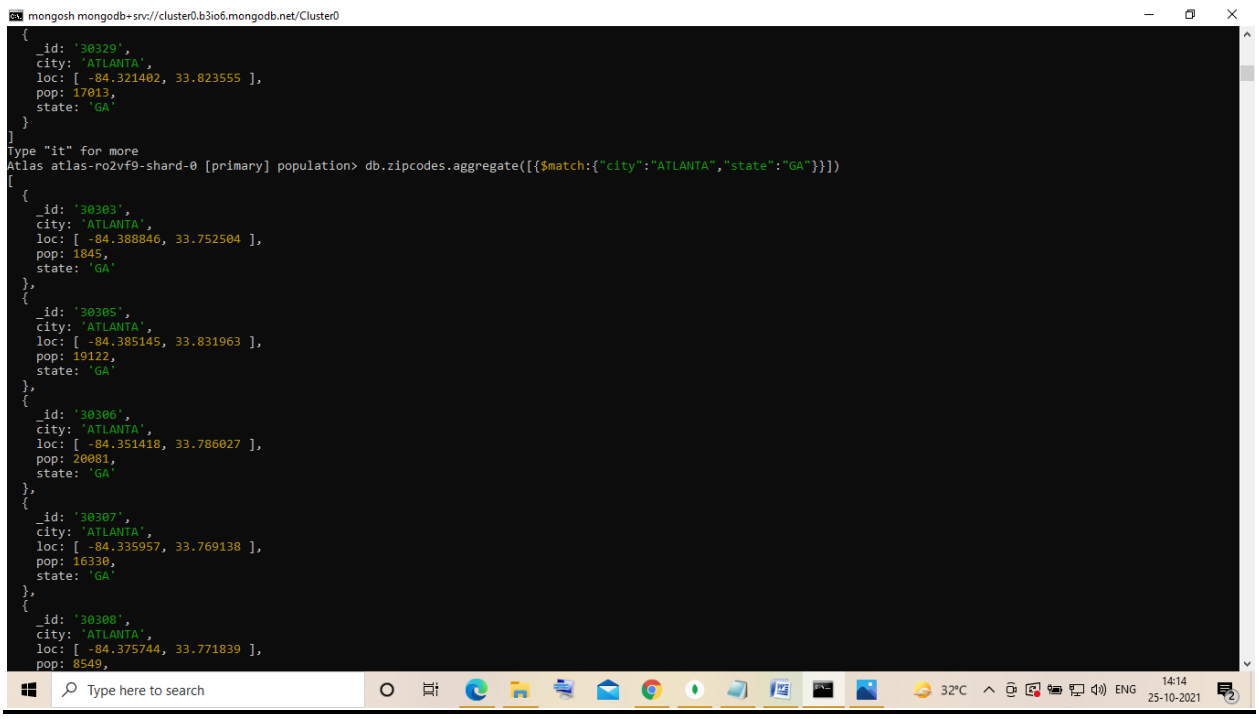
show collections

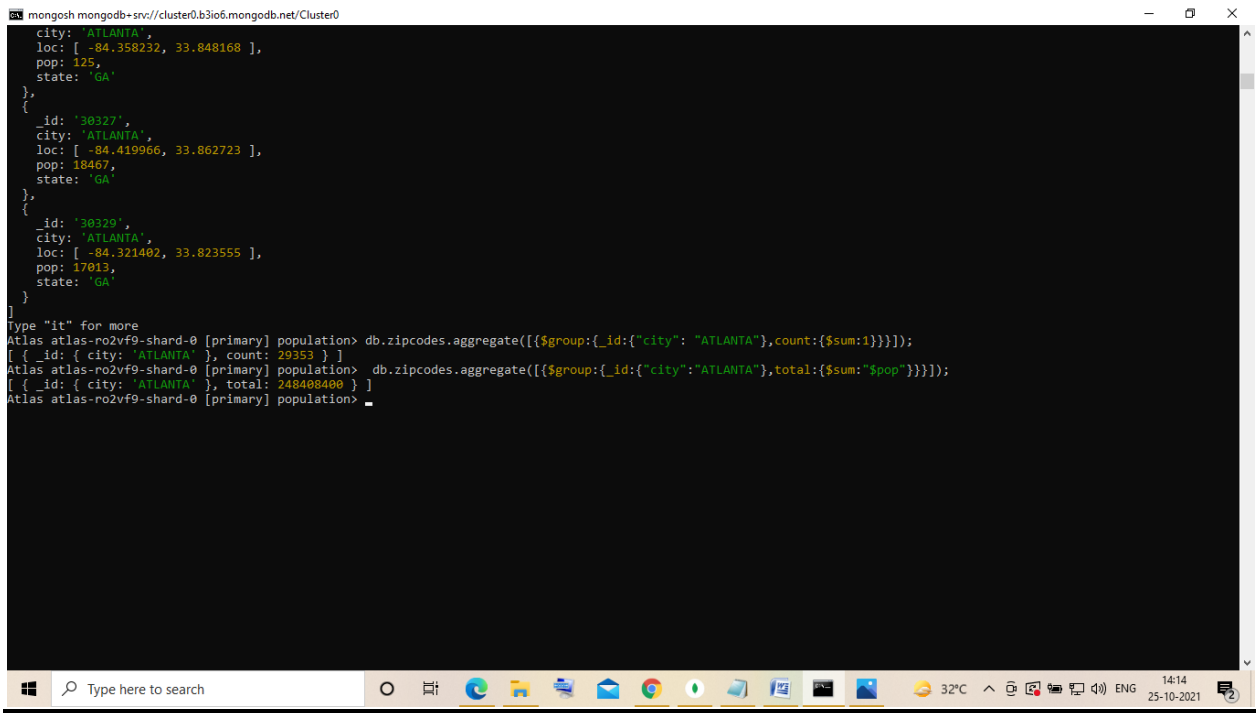
db.zipcodes.find().pretty();



Atlanta Population:-







<u>Use db.zipcodes.find() to filter results to only the results where city is ATLANTA and state is GA :-</u>

db.zipcodes.find({"city":"ATLANTA","state":"GA"});

<u>Use db.zipcodes.aggregate with $match to do the same as above :-</u>

db.zipcodes.aggregate([{$match:{"city":"ATLANTA","state":"GA"}}]);

<u>Use $group to count the number of zip codes in Atlanta :-</u>

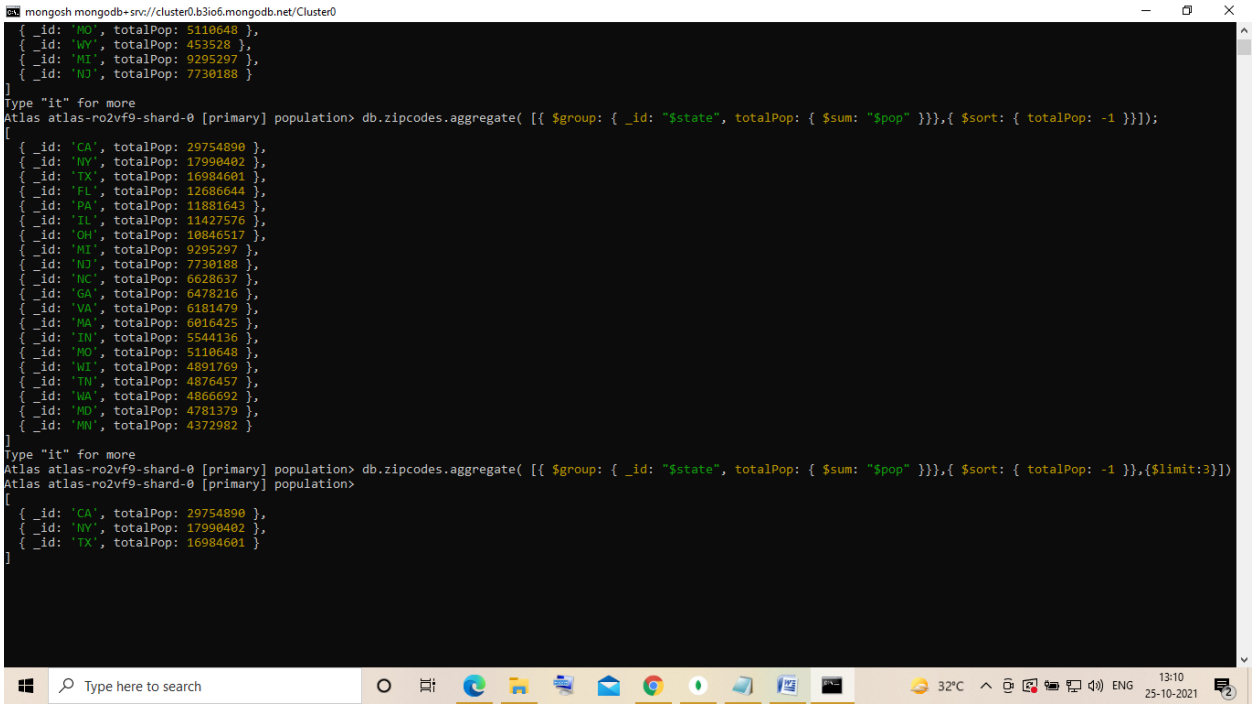
db.zipcodes.aggregate([{$group:{_id:{"city": "ATLANTA"},count:{$sum:1}}}]);

<u>Use $group to find the total population in Atlanta:-</u>

db.zipcodes.aggregate([{$group:{_id:{"city":"ATLANTA"},total:{$sum:"$pop"}}}]);

db.zipcodes.aggregate([{$match:{"city":"ATLANTA"}},{$group:{_id:"$city",total:{$sum:"$pop"}}}]);

Populations By State:-



Use aggregate to calculate the total population for each state :-

db.zipcodes.aggregate([{ $group: { _id: "$state", totalPop: { $sum: "$pop" }}}]);

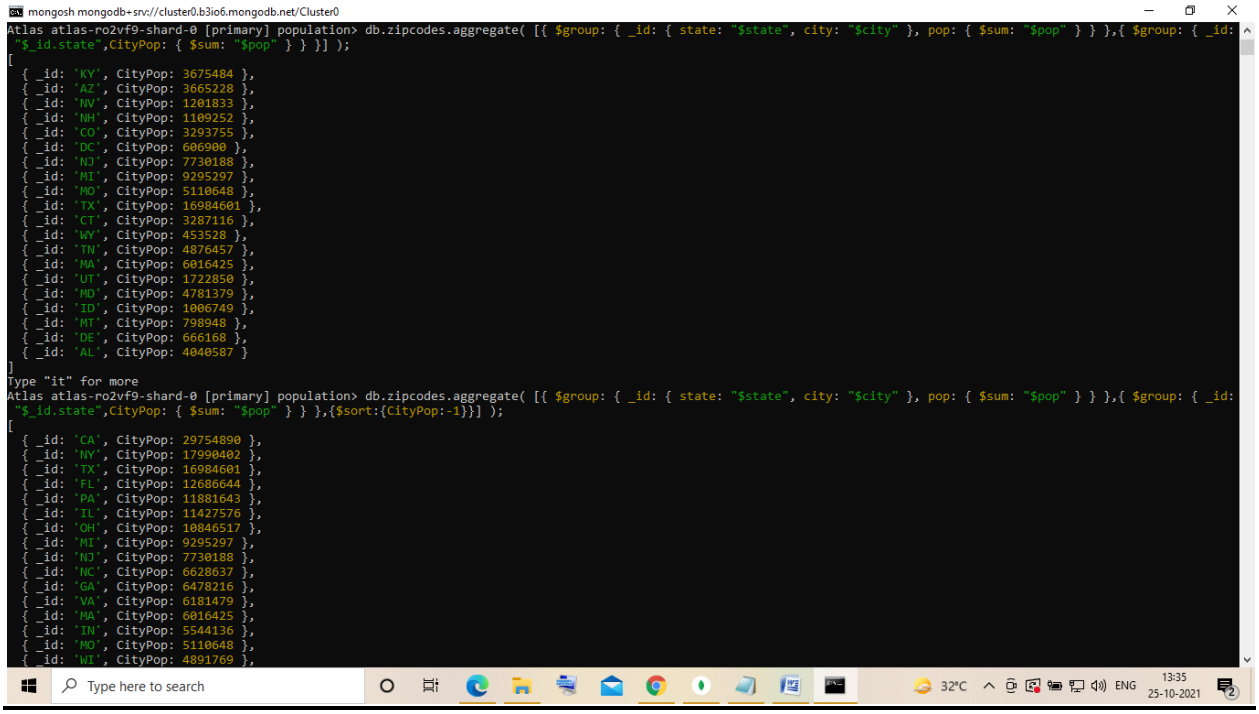
Sort the results by population, highest first:-

db.zipcodes.aggregate([{ $group: { _id: "$state", totalPop: { $sum: "$pop" }}},{ $sort: { totalPop: -1 }}]);

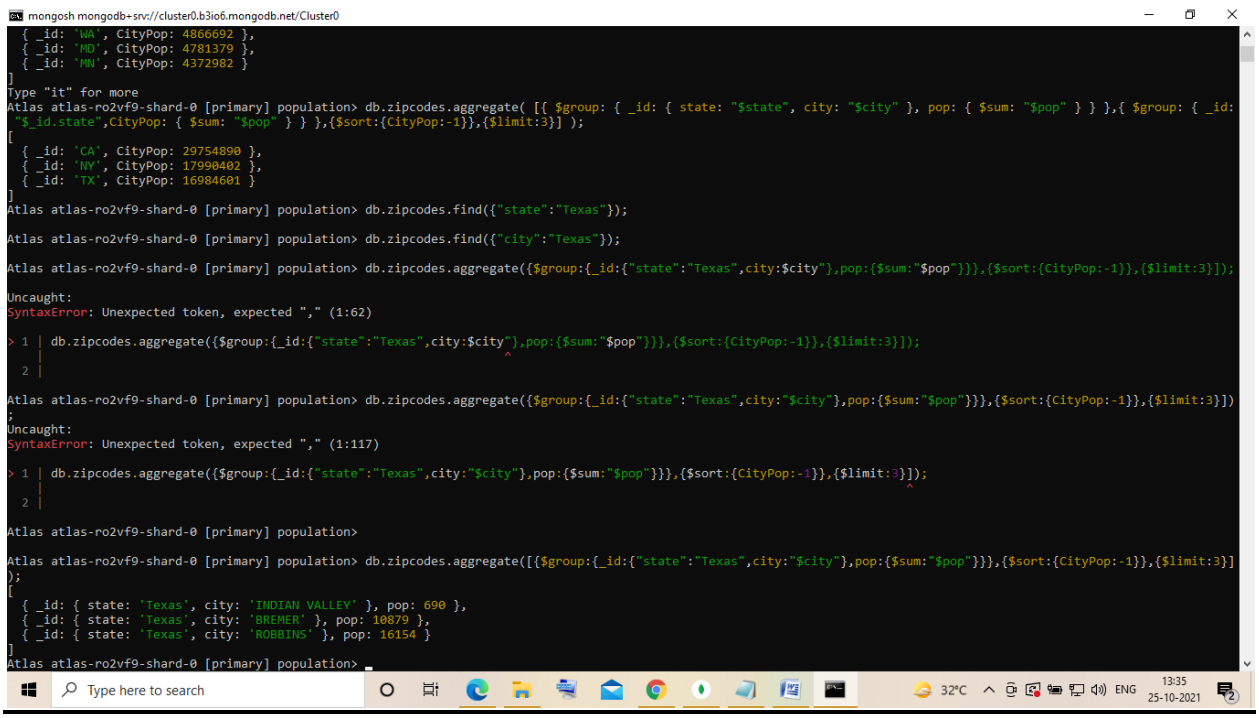
Limit the results to just the first 3 results. What are the top 3 states in population? :-

db.zipcodes.aggregate([{ $group: { _id: "$state", totalPop: { $sum: "$pop" }}},{ $sort: { totalPop: -1 }},{$limit:3}]);

Populations by City :-



```
mongosh mongodb+srv://cluster0.b3io6.mongodb.net/Cluster0
Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.aggregate( [{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id:
 "$_id.state",CityPop: { $sum: "$pop" } } }] );
[
  { _id: 'KY', CityPop: 3675484 },
  { _id: 'AZ', CityPop: 3665228 },
  { _id: 'NV', CityPop: 1201833 },
  { _id: 'NH', CityPop: 1109252 },
  { _id: 'CO', CityPop: 3293755 },
  { _id: 'DC', CityPop: 606900 },
  { _id: 'NJ', CityPop: 7730188 },
  { _id: 'MI', CityPop: 9295297 },
  { _id: 'MO', CityPop: 5110648 },
  { _id: 'TX', CityPop: 16984601 },
  { _id: 'CT', CityPop: 3287116 },
  { _id: 'WY', CityPop: 453528 },
  { _id: 'TN', CityPop: 4876457 },
  { _id: 'MA', CityPop: 6016425 },
  { _id: 'UT', CityPop: 1722850 },
  { _id: 'MD', CityPop: 4781379 },
  { _id: 'ID', CityPop: 1006749 },
  { _id: 'MT', CityPop: 798948 },
  { _id: 'DE', CityPop: 666168 },
  { _id: 'AL', CityPop: 4040587 }
]
Type "it" for more
Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.aggregate( [{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id:
 "$_id.state",CityPop: { $sum: "$pop" } } },{$sort:{CityPop:-1}}] );
[
  { _id: 'CA', CityPop: 29754890 },
  { _id: 'NY', CityPop: 17990402 },
  { _id: 'TX', CityPop: 16984601 },
  { _id: 'FL', CityPop: 12686644 },
  { _id: 'PA', CityPop: 11881643 },
  { _id: 'IL', CityPop: 11427576 },
  { _id: 'OH', CityPop: 10846517 },
  { _id: 'MI', CityPop: 9295297 },
  { _id: 'NJ', CityPop: 7730188 },
  { _id: 'NC', CityPop: 6628637 },
  { _id: 'GA', CityPop: 6478216 },
  { _id: 'VA', CityPop: 6181479 },
  { _id: 'MA', CityPop: 6016425 },
  { _id: 'IN', CityPop: 5544136 },
  { _id: 'MO', CityPop: 5110648 },
  { _id: 'WI', CityPop: 4891769 },
```



```
  { _id: 'WA', CityPop: 4866692 },
  { _id: 'MD', CityPop: 4781379 },
  { _id: 'MN', CityPop: 4372982 }
]
Type "it" for more
Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.aggregate( [{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id:
 "$_id.state",CityPop: { $sum: "$pop" } } },{$sort:{CityPop:-1}},{$limit:3}] );
[
  { _id: 'CA', CityPop: 29754890 },
  { _id: 'NY', CityPop: 17990402 },
  { _id: 'TX', CityPop: 16984601 }
]
Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.find({"state":"Texas"});

Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.find({"city":"Texas"});

Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.aggregate({$group:{_id:{"state":"Texas",city:$city"},pop:{$sum:"$pop"}}},{$sort:{CityPop:-1}},{$limit:3}]);

Uncaught:
SyntaxError: Unexpected token, expected "," (1:62)

> 1 | db.zipcodes.aggregate({$group:{_id:{"state":"Texas",city:$city"},pop:{$sum:"$pop"}}},{$sort:{CityPop:-1}},{$limit:3}]);
    |                                                              ^
  2 |

Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.aggregate({$group:{_id:{"state":"Texas",city:"$city"},pop:{$sum:"$pop"}}},{$sort:{CityPop:-1}},{$limit:3}])
;
Uncaught:
SyntaxError: Unexpected token, expected "," (1:117)

> 1 | db.zipcodes.aggregate({$group:{_id:{"state":"Texas",city:"$city"},pop:{$sum:"$pop"}}},{$sort:{CityPop:-1}},{$limit:3}]);
    |                                                                                                             ^
  2 |

Atlas atlas-ro2vf9-shard-0 [primary] population>

Atlas atlas-ro2vf9-shard-0 [primary] population> db.zipcodes.aggregate([{$group:{_id:{"state":"Texas",city:"$city"},pop:{$sum:"$pop"}}},{$sort:{CityPop:-1}},{$limit:3}]
);
[
  { _id: { state: 'Texas', city: 'INDIAN VALLEY' }, pop: 690 },
  { _id: { state: 'Texas', city: 'BREMER' }, pop: 10879 },
  { _id: { state: 'Texas', city: 'ROBBINS' }, pop: 16154 }
]
Atlas atlas-ro2vf9-shard-0 [primary] population>
```

Use aggregate to calculate the total population for each city (you have to use city/state combination). You can use a combination for the _id of the $group: { city: '$city', state: '$state' }:-

```
db.zipcodes.aggregate( [{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state",CityPop: { $sum: "$pop" } } }] );
```

Sort the results by population, highest first:-

```
db.zipcodes.aggregate( [{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state",CityPop: { $sum: "$pop" } } },{$sort:{CityPop:-1}}] );
```

Limit the results to just the first 3 results. What are the top 3 cities in population? :-

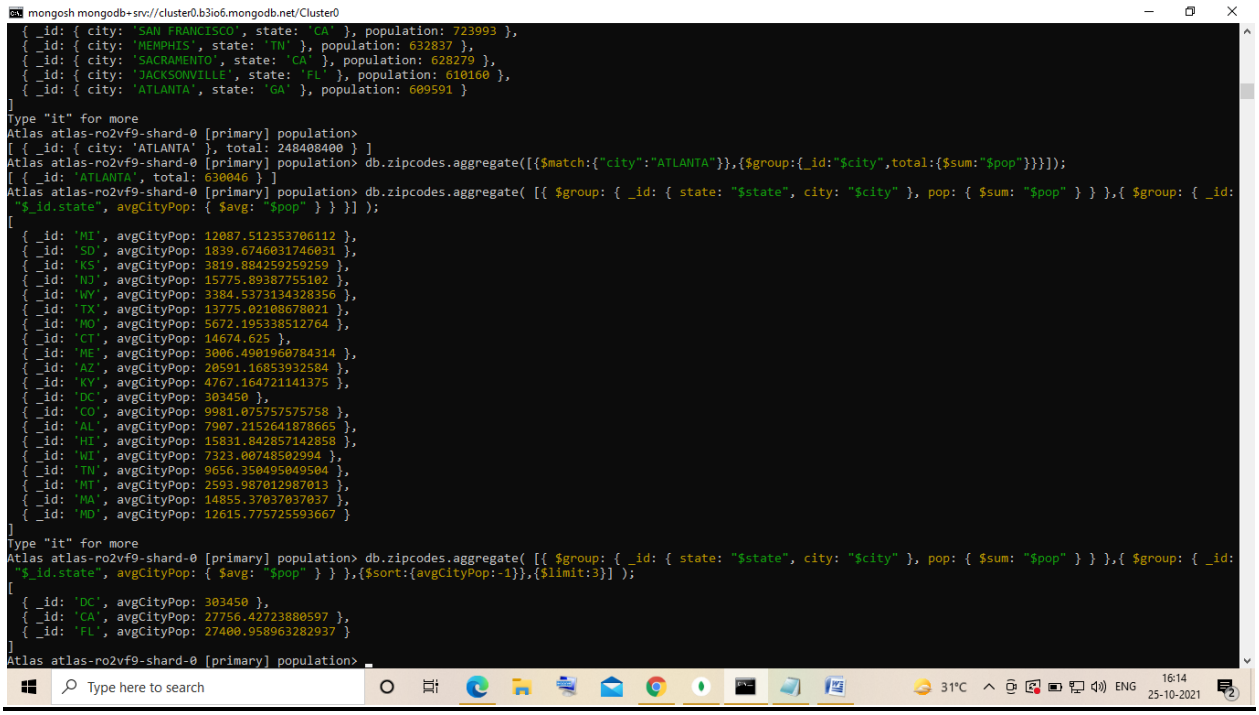
```
db.zipcodes.aggregate( [{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state",CityPop: { $sum: "$pop" } } },{$sort:{CityPop:-1}},{$limit:3}] );
```

What are the top 3 cities in population in Texas? :-

```
db.zipcodes.aggregate([{$match:{state:"TX"}},{$group:{_id:"$city",pop:{$max:"$pop"}}},{$sort:{pop:-1}},{$limit:3}]);
```

```
db.zipcodes.aggregate([{$group:{_id:{state:"Texas",city:"$city"},pop:{$max:"$pop"}}},{$sort:{pop:-1}},{$limit:3}]);
```

Bonus :-

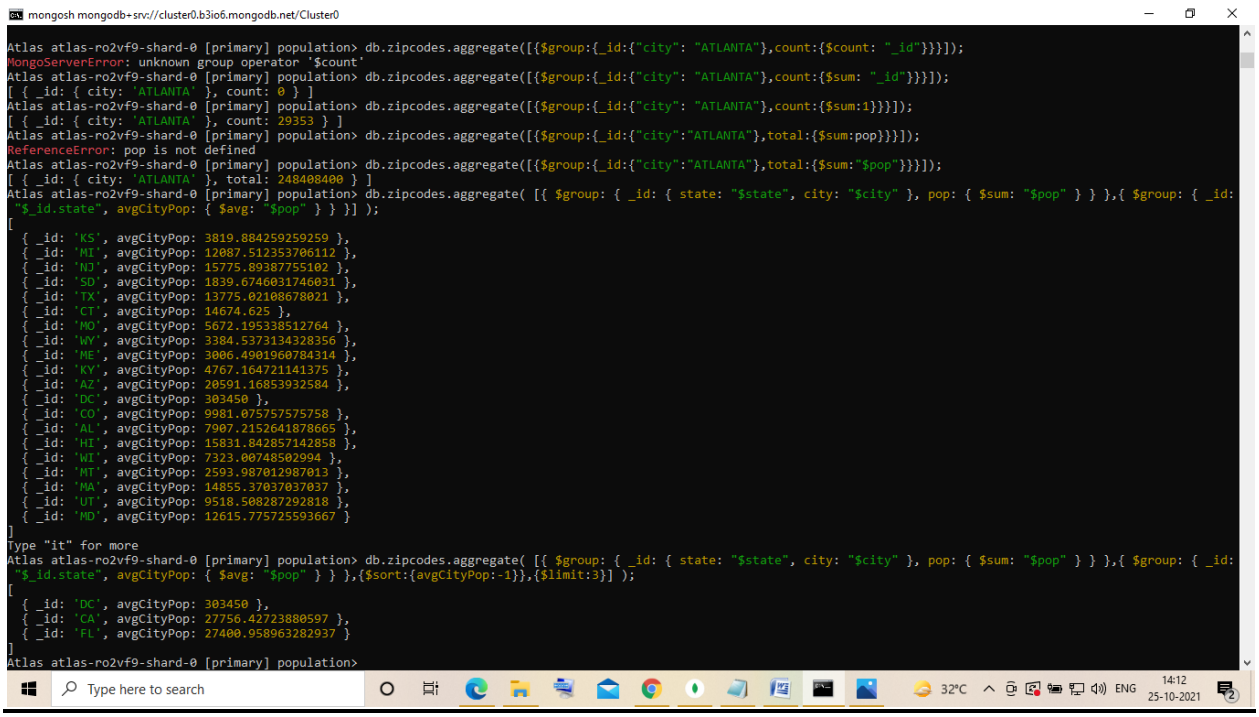


1. Write a query to get the average city population for each state.

db.zipcodes.aggregate([{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state", avgCityPop: { $avg: "$pop" } } }]);

2. What are the top 3 states in terms of average city population.

db.zipcodes.aggregate([{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state", avgCityPop: { $avg: "$pop" } } },{$sort:{avgCityPop:-1}},{$limit:3}]);

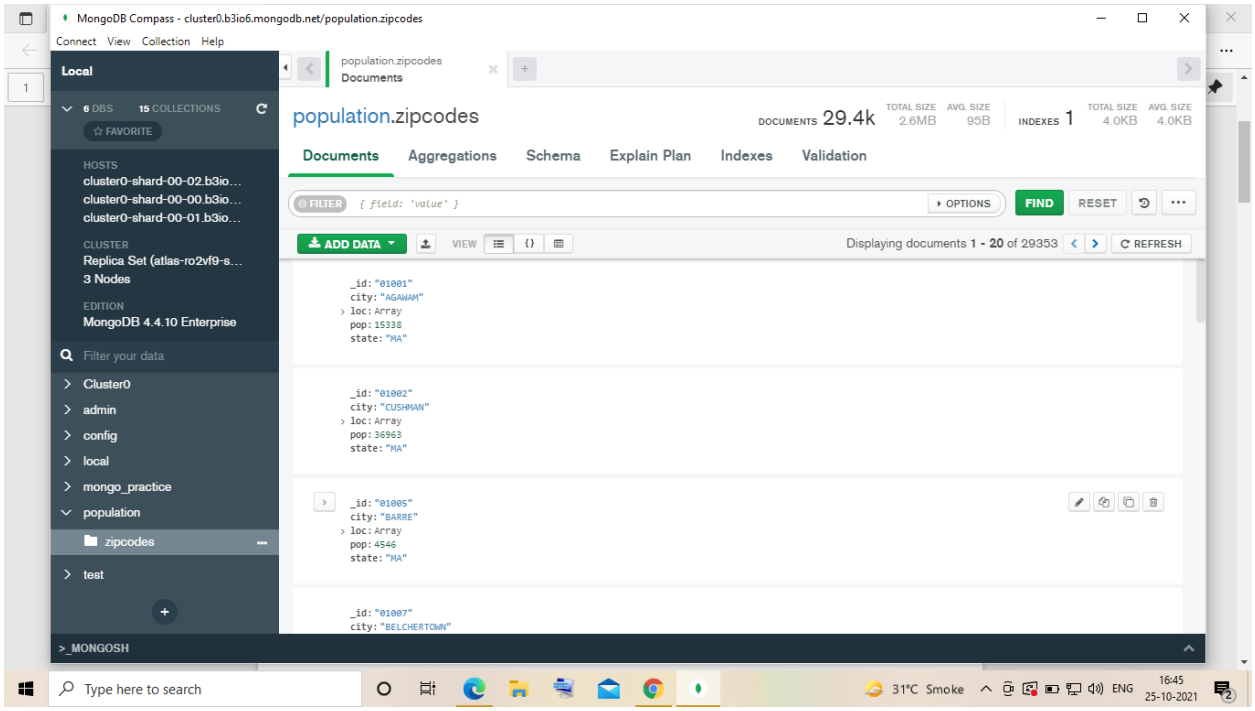


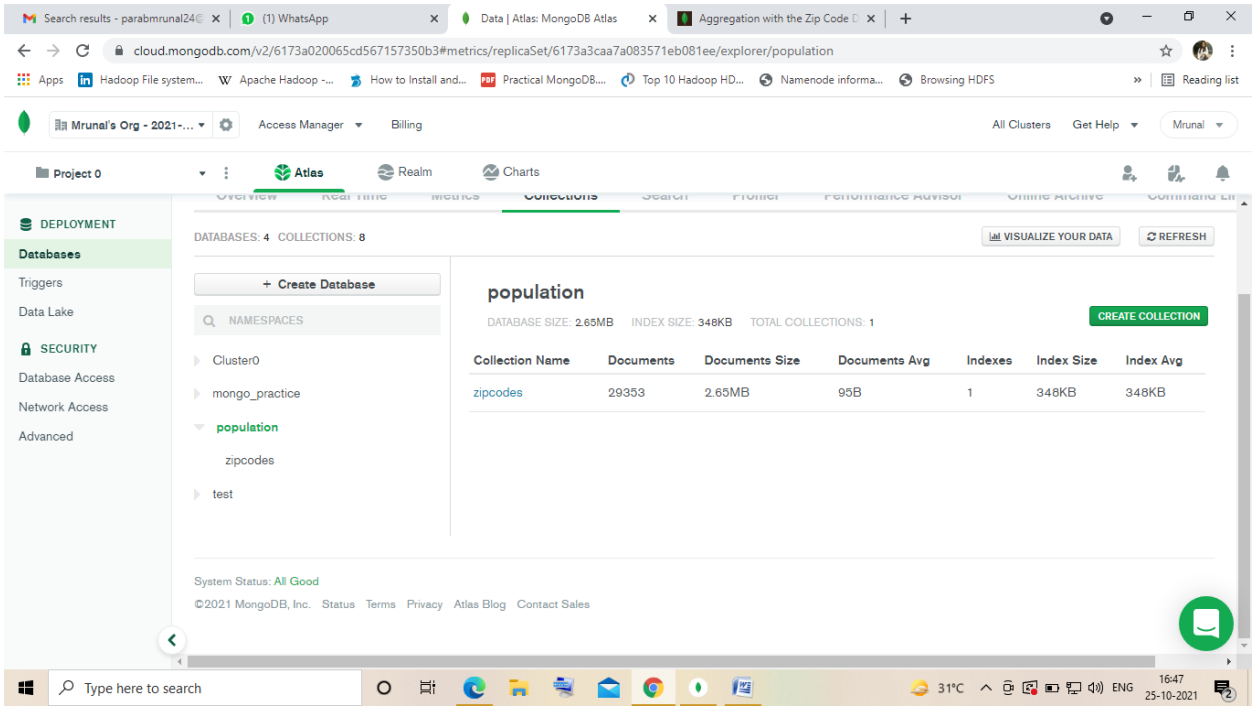
1. Write a query to get the average city population for each state.

db.zipcodes.aggregate([{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state", avgCityPop: { $avg: "$pop" } } }]);

2. What are the top 3 states in terms of average city population.

db.zipcodes.aggregate([{ $group: { _id: { state: "$state", city: "$city" }, pop: { $sum: "$pop" } } },{ $group: { _id: "$_id.state", avgCityPop: { $avg: "$pop" } } },{$sort:{avgCityPop:-1}},{$limit:3}]);


MongoDB Compass - cluster0.b3io6.mongodb.net/population.zipcodes
Connect View Collection Help
Local
6 DBS 15 COLLECTIONS
FAVORITE
HOSTS
cluster0-shard-00-02.b3io...
cluster0-shard-00-00.b3io...
cluster0-shard-00-01.b3io...
CLUSTER
Replica Set (atlas-ro2vf9-s...
3 Nodes
EDITION
MongoDB 4.4.10 Enterprise
Filter your data
Cluster0
admin
config
local
mongo_practice
population
zipcodes
test
>_MONGOSH
population.zipcodes
Documents
population.zipcodes
DOCUMENTS 29.4k
TOTAL SIZE 2.6MB
AVG. SIZE 95B
INDEXES 1
TOTAL SIZE 4.0KB
AVG. SIZE 4.0KB
Documents
Aggregations
Schema
Explain Plan
Indexes
Validation
FILTER { field: 'value' }
OPTIONS
FIND
RESET
ADD DATA
VIEW
Displaying documents 1 - 20 of 29353
REFRESH
_id: "01001"
city: "AGAWAM"
loc: Array
pop: 15338
state: "MA"
_id: "01002"
city: "CUSHMAN"
loc: Array
pop: 36963
state: "MA"
_id: "01005"
city: "BARRE"
loc: Array
pop: 4546
state: "MA"
_id: "01007"
city: "BELCHERTOWN"
Type here to search
31°C Smoke
ENG
16:45
25-10-2021


Search results - parabmrunal24
(1) WhatsApp
Data | Atlas: MongoDB Atlas
Aggregation with the Zip Code
cloud.mongodb.com/v2/6173a020065cd567157350b3#metrics/replicaSet/6173a3caa7a083571eb081ee/explorer/population
Apps
Hadoop File system...
Apache Hadoop -...
How to Install and...
Practical MongoDB...
Top 10 Hadoop HD...
Namenode informa...
Browsing HDFS
Reading list
Mrunal's Org - 2021-...
Access Manager
Billing
All Clusters
Get Help
Mrunal
Project 0
Atlas
Realm
Charts
DEPLOYMENT
Databases
Triggers
Data Lake
SECURITY
Database Access
Network Access
Advanced
DATABASES: 4 COLLECTIONS: 8
VISUALIZE YOUR DATA
REFRESH
+ Create Database
NAMESPACES
Cluster0
mongo_practice
population
zipcodes
test
population
DATABASE SIZE: 2.65MB INDEX SIZE: 348KB TOTAL COLLECTIONS: 1
CREATE COLLECTION
Collection Name
Documents
Documents Size
Documents Avg
Indexes
Index Size
Index Avg
zipcodes
29353
2.65MB
95B
1
348KB
348KB
System Status: All Good
©2021 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales
Type here to search
31°C
ENG
16:47
25-10-2021