**MongoDB -Aggregation Exercises**

Import the zips.json file into your MongoDB. Database name is "population" and collection name is "zipcodes".

mongoimport --db population --collection zipcodes --file zips.json

Atlanta Population

1. use db.zipcodes.find() to filter results to only the results where city is ATLANTA and state is GA.

ANS. db.zipcodes.find({“citi”:”ATLANTA”,”state”:”GA”})

1. use db.zipcodes.aggregate with $match to do the same as above.

Ans. db.zipcodes.aggregate([

... {$match:{"city":"ATLANTA","state":"GA"}}

... ])

1. use $group to count the number of zip codes in Atlanta.

Ans. db.zipcodes.aggregate([

... {$group:{\_id:{'city':'ATLANTA'},count:{$sum:1}}}

... ])

1. use $group to find the total population in Atlanta

ans. db.zipcodes.aggregate([

... {$group:{\_id:{'city':'ATLANTA'},population:{$sum:'$pop'}}}

... ])

Populations By State

1. use aggregate to calculate the total population for each state

ans. db.zipcodes.aggregate([

... { $group : { \_id : {'state':"$state"},totalPop: { $sum: "$pop" }}}

... ... ])

1. sort the results by population, highest first

ans.

db.zipcodes.aggregate([

... ... {$sort:{"pop":-1}}

])

\_s

1. limit the results to just the first 3 results. What are the top 3 states in population?

Ans. db.zipcodes.aggregate([

... {$sort:{"pop":-1}},

... ... {$limit:3}

... ... ])

{ "\_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" }

Populations by City

1. 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' }

Ans. db.zipcodes.aggregate([

{$group:{\_id:{city:’$city’,state:’$state’},population:{$sum:’$pop’}}}

])

2. sort the results by population, highest first

ans.

db.zipcodes.aggregate([

... ... {$sort:{"pop":-1}}

])

3.limit the results to just the first 3 results. What are the top 3 cities in population?

Ans. db.zipcodes.aggregate([

... {$sort:{"pop":-1}},

... ... {$limit:3}

... ... ])

{ "\_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" }

1. What are the top 3 cities in population in Texas?

Ans. db.zipcodes.aggregate([

... ... ... {$group:{\_id:{'city':"$city",'pop':"TEXAS"}}},

... ... {$sort:{"pop":-1}},

... {$limit:3}

])

Bonus

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

Ans. db.zipcodes.aggregate([

... { $group: { \_id: { city: "$city", state: "$state"}, pop: { $sum: "$pop" } } },

... { $group: { \_id: "$\_id.state", avgCityPop: { $avg: "$pop" } } }

... ])

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

Ans. db.zipcodes.aggregate([

... ... { $group: { \_id: { city: "$city", state: "$state"}, pop: { $sum: "$pop" } } },

... ... { $group: { \_id: "$\_id.state", avgCityPop: { $avg: "$pop" } } },

... {$sort:{"avgCityPop":-1}},

... {$limit:3}

... ])

{ "\_id" : "DC", "avgCityPop" : 303450 }

{ "\_id" : "CA", "avgCityPop" : 27756.42723880597 }

{ "\_id" : "FL", "avgCityPop" : 27400.958963282937 }