# MongoDB -Aggregation Exercises

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

use population

db.CreateCollection(“zipcodes”)

steps to import file-

{click on new collection-select import data}

importfile-(JSON(input file type)-select file- click on import

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.

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

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

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

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

db.zipcodes.aggregate ({$match:{city: 'ATLANTA'}},{$group:{\_id: '$city',zips: {$sum: 1}}},{$sort:{zips: -1}})

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

db.zipcodes.aggregate ({$match:{city: 'ATLANTA'}},{$group:{\_id: {city: '$city'},population: {$sum: '$pop'}}})

ans-{ "\_id" : { "city" : "ATLANTA" }, "population" : 630046 }

# Populations By State

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

db.zipcodes.aggregate ( {$group: { \_id: '$state', population: {$sum: '$pop'} } } )

1. sort the results by population, highest first

db.zipcodes.aggregate ( {$group: { \_id: {city: '$city', state: '$state',population:'$pop'}} }, {$sort: {population: -1} } )

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

db.zipcodes.aggregate ({$group:{\_id: {state: '$state',population:'$pop'}}}, { $limit : 3 },{$sort:{population:-1}})

# 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' }

db.zipcodes.aggregate ({$group:{\_id: {city: '$city', state: '$state'},population: {$sum: '$pop'}}})

1. sort the results by population, highest first

db.zipcodes.aggregate ({$group:{\_id: {city: '$city', state: '$state'},population: {$sum: '$pop'}}},{$sort:{population: -1}})

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

db.zipcodes.aggregate ({$group:{\_id: {city: '$city',population:'$pop'}}}, { $limit : 3 },{$sort:{population:-1}})

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

db.zipcodes.aggregate([{$group:{\_id:{city:"$city",state:"$Texas"},totalpop:{$sum:"$pop"}}},{$sort{totalpop:-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" } } }] )

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

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