Lab 9 (2%)

MongoDB: Aggregation

Learning outcomes

Upon successful completion of this workshop, you will have demonstrated the abilities to:

1. Use the aggregation framework to transform and combine documents in a collection.
2. Run simple aggregation commands such as count, distinct and group.

Done by : Mansoor Ahmad Zafar

Data:

Restaurant-data.json

Aggregation in MongoDB

Write MongoDB queries to answer each of the following questions.

1. Show the number of restaurants per cuisine type. Hint: Use the aggregate() function and the $group operator.

db.restaurant.aggregate({

"$group" :

{

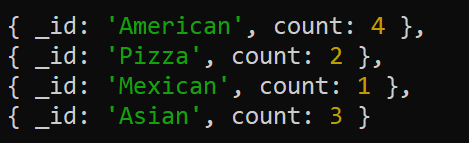
"\_id" : "$cuisine\_type",

"count" : {"$sum" : 1}

}

})

output:



1. Show the number of restaurants in each neighborhood. Sort the restaurants in descending order of their count and limit the results to the first 2 result documents.

output:



db.restaurant.aggregate(

{"$group" :

{

"\_id" : "$neighborhood",

"count" : {"$sum" : 1}

}},

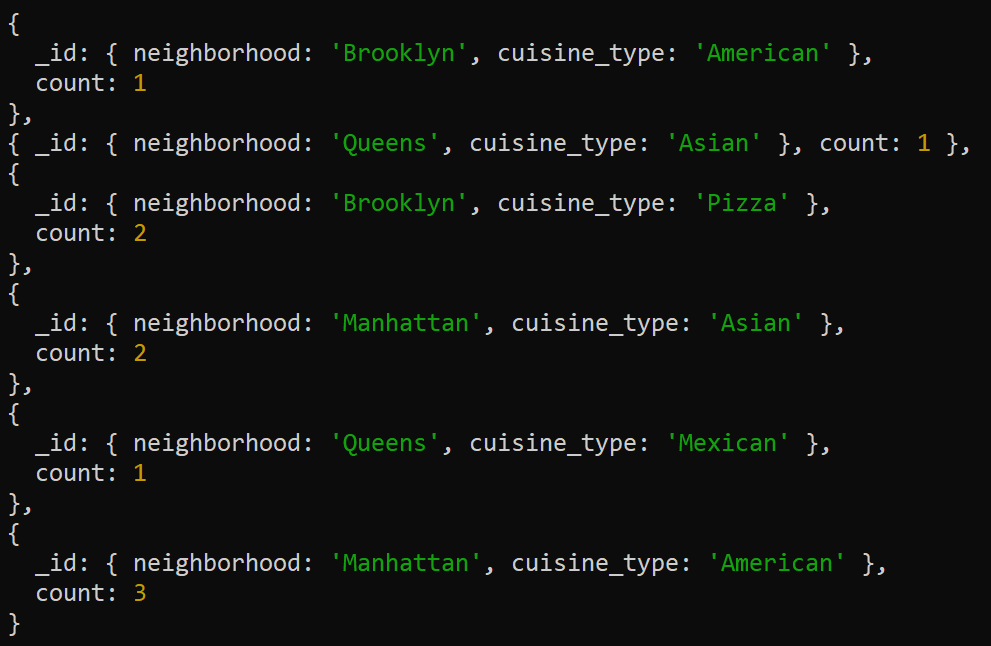
{"$sort" : {"count" : -1}},

{"$limit" : 2}

)

1. Show the number of restaurants per neighborhood and cuisine type. Use the aggregate() function and the $group operator.

output:



db.restaurant.aggregate(

{"$group" :

{

"\_id" : {"neighborhood" : "$neighborhood", "cuisine\_type" : "$cuisine\_type"},

"count" : {"$sum" : 1}

}

}

)

1. Concatenate the restaurant name and the cuisine type into one string literal as restauType key. Hide \_id. Hint: Use the aggregate() function, $project operator and $concat operator.

db.restaurant.aggregate(

{"$project" :

{

"restauType" : { "$concat" : ["$name", " - ", "$cuisine\_type"]}

}

}

)

output:



**count()**

1. calculate the number of documents in the collection. Output: 10

db.restaurant.count()

**$unwind**

1. Unwind the restaurants collection and show the number of reviews per restaurant.

db.restaurant.aggregate(

{"$unwind" : "$reviews"},

{"$group" :

{

"\_id" : "$name",

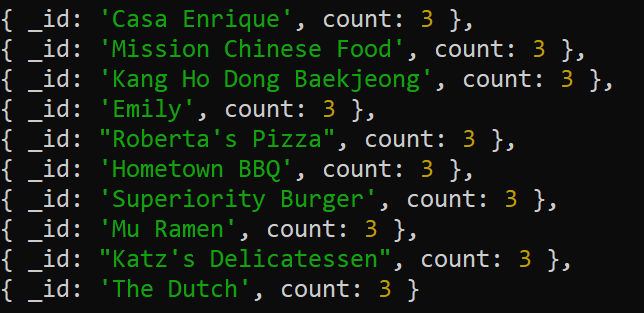
"count" : {"$sum" : 1}

}

}

)

output:

****

1. Unwind the restaurants collection and show the number of reviews per user name.

db.restaurant.aggregate

(

{"$unwind" : "$reviews"},

{

"$group" :

{

"\_id" : "$reviews.name",

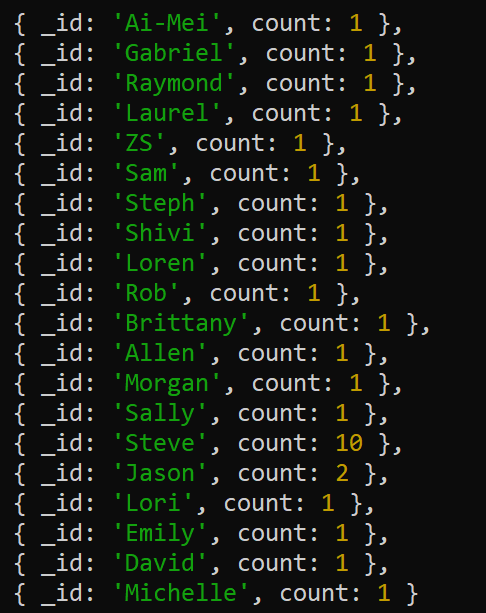
"count" : {"$sum" : 1}

}

}

)

Output:



1. Unwind the restaurants collection and show the average rating per user name.

db.restaurant.aggregate

(

{"$unwind" : "$reviews"},

{

"$group" :

{

"\_id" : "$reviews.name",

"avgRating" : {"$avg" : "$reviews.rating"}

}

}

)



SUBMISSION

Submit lab9\_Groupxx.doc file on BB.

Also submit lab9\_Groupxx.txt with question number and the command in text.

4 marks per question, total: 32 marks.