Homework 6
CSCI 3287
Brandon Glandt
4/19/2020

1. Section 1

```
> use new_db
switched to db new_db
> db.dropDatabase()
{ "ok" : 1 }
> db.createCollection("test_collection")
{ "ok" : 1 }
> db.test_collection.drop()
true
> db.test_collection.insert({
... title: "Mongo Db practice",
... description: "this is my first MongoDB document"
... })
WriteResult({ "nInserted" : 1 })
> db.test_collection.find().pretty()
{
        "_id" : ObjectId("5e94ba0fc87879e65ac6340e"),
        "title" : "Mongo Db practice",
        "description" : "this is my first MongoDB document"
}
> db.test_collection.find()
{ "_id" : ObjectId("5e94ba0fc87879e65ac6340e"), "title" : "Mongo Db practice", "description" : "this is my first MongoDB document" }
> db.test_collection.update({'title':'MongoDB practice'},{$set:{ 'title':'Updated MongoDB practice'}})
WriteResult({ "nMatched" : 0, "nUpserted" : 0, "nModified" : 0 })
> db.test_collection.remove({status : "P"},1)
WriteResult({ "nRemoved" : 0 })
> db.test_collection.remove({status : "P" })
WriteResult({ "nRemoved" : 0 })
>
```

2. Section 2
   a. How many restaurants are there in this collection?

```
> db.restaurants.find().count()
25359
>
```

   b. List in alphabetical order each different (distinct) cuisine represented in this collection.

```
[> db.restaurants.distinct("cuisine")
[
        "Afghan",
        "African",
        "American",
        "Armenian",
        "Asian",
        "Australian",
        "Bagels/Pretzels",
        "Bakery",
        "Bangladeshi",
        "Barbecue",
        "Bottled beverages, including water, sodas, juices, etc.",
        "Brazilian",
        "CafÃ©/Coffee/Tea",
        "Café/Coffee/Tea",
        "Cajun",
        "Californian",
        "Caribbean",
        "Chicken",
        "Chilean",
        "Chinese",
        "Chinese/Cuban",
        "Chinese/Japanese",
        "Continental",
        "Creole",
        "Creole/Cajun",
        "Czech",
        "Delicatessen",
        "Donuts",
        "Eastern European",
        "Egyptian",
        "English",
        "Ethiopian",
        "Filipino",
        "French",
        "Fruits/Vegetables",
        "German",
        "Greek",
        "Hamburgers",
        "Hawaiian",
        "Hotdogs",
        "Hotdogs/Pretzels",
        "Ice Cream, Gelato, Yogurt, Ices",
        "Indian",
        "Indonesian",
        "Iranian",
        "Irish",
        "Italian",
        "Japanese",
        "Jewish/Kosher",
```

c. Return the name of all restaurants within the zipcode 11215 which serve Indian cuisine. Return only the names of the restaurants.

```
> db.restaurants.find({$and:[{"cuisine":"Indian"},{"address.zipcode":"11215"}]},
{_id:0,name:1});
{ "name" : "Kinara Indian Restaurant" }
{ "name" : "Baluchi'S" }
{ "name" : "Kanan Indian Restaurant" }
{ "name" : "New Aarpan" }
{ "name" : "Indian Spice" }
>
```

d. Return the count of restaurants in the Bronx that serve either Chinese or American Food

```
> db.restaurants.find( { "borough" : "Bronx", $or : [ {"cuisine" : "American"},
{"cuisine" : "Chinese"} ]}).count();
734
>
```

e. Return a list of restaurants (names) which have the string “Food” in their name.

```
> db.restaurants.find({"name":/Food/},{name:1,_id:0}).pretty()
{ "name" : "Wilken'S Fine Food" }
{ "name" : "Seuda Foods" }
{ "name" : "Glorious Food" }
{ "name" : "American Museum Of Natural History Food Court" }
{ "name" : "Pax Wholesome Foods" }
{ "name" : "Pax Wholesome Foods" }
{ "name" : "Fordham Fried Chicken & Sea Food" }
{ "name" : "Downtown Bakery Ii Mexican Food" }
{ "name" : "Columbus Gourmet Food" }
{ "name" : "Food For Thought Library Cafe" }
{ "name" : "Food Mart Deli" }
{ "name" : "Food Merchants" }
{ "name" : "Metropolitan Food Cafe Of Brooklyn College" }
{ "name" : "Food Fair Deli & Pizza" }
{ "name" : "Tasty Fast Food" }
{ "name" : "The Food Hut" }
{ "name" : "Food For Thought Catered Events" }
{ "name" : "Tandoori Food & Bakery" }
{ "name" : "Snack Bar (Located Between A-B Between Fancy Food And Masters)" }
{ "name" : "Reliable Food" }
Type "it" for more
>
```

f. Return a list of boroughs ranked by the number of Italian restaurants in the borough. That is, for each borough, find how many restaurants serve Italian cuisine and print the borough and the number of such restaurants sorted descending by this number. (HINT: use the aggregate method, and use a $group and a $sum.)

```
[> db.restaurants.aggregate([{$match:{cuisine:"Italian"}},{$group:{_id:'$borough'}
,amount: {$sum:1}}},{$sort:{amount:-1}}])
{ "_id" : "Manhattan", "amount" : 621 }
{ "_id" : "Brooklyn", "amount" : 192 }
{ "_id" : "Queens", "amount" : 131 }
{ "_id" : "Staten Island", "amount" : 73 }
{ "_id" : "Bronx", "amount" : 52 }
>
```

g. Find the top 5 French restaurants in Manhattan that have the highest total score. Return for each restaurant the restaurant's name and the total score. (HINT: use the aggregate method with $unwind to parse out the scores array, followed by a $group and a $sum.)

```
[> db.restaurants.aggregate([{$match:{"cuisine":"French","borough":"Manhattan"}},
{$unwind:"$grades"},{$group:{_id:'$name',totalScore:{$sum:"$grades.score"}}},{$s
ort:{totalScore:-1}},{$limit:5}])
{ "_id" : "Josephine", "totalScore" : 143 }
{ "_id" : "Petite Abeille", "totalScore" : 142 }
{ "_id" : "Financier Patisserie", "totalScore" : 133 }
{ "_id" : "Cafe Un Deux Trois", "totalScore" : 132 }
{ "_id" : "Tout Va Bien", "totalScore" : 129 }
>
```