John Furlong
CSCI 3287
Extra Credit Assignment #3
December 18, 2019

5 New MongoDB Queries

1. Write a mongoDB query to find the number of restaurants in the Brooklyn borough that serve either Italian or French cuisine.
    a. (Utilizes $or)

```
> db.restaurants.find(
... { borough : "Brooklyn",
...     $or :
...        [
...          { cuisine : "Itaian"},
...          { cuisine : "French"}
...        ]
... }).count()
54
```

2. Write a mongoDB query that returns the number of restaurants in either Queens, Brooklyn, or Staten Island that do not serve German cuisine.
    a. (Utilizes $and as well as $in)

```
> db.restaurants.find(
...   { $and :
...      [
...         { borough : { $in : ["Staten Island", "Brooklyn", "Queens"] } },
...         { cuisine : { $ne : "German"} }
...      ]
...   } ).count()
12696
>
```

3. Write a mongoDB query that returns the name and the count of the borough that has the most “Subway” restaurants.
    a. (Utilize a $group on the count of a certain restaurant $name)

```
> db.restaurants.aggregate([
... { $match : {name : "Subway"} },
... { $group : {
...       _id : "$borough",
...       "Number of Subways" : {$sum:1} } },
... { $sort : {"Number of Subways" : -1} },
... { $limit : 1} ] )
{ "_id" : "Manhattan", "Number of Subways" : 152 }
>
```

4. Write a mongoDB query to find the names of the 3 most-eastward restaurants that serve American cuisine. (Hint : East-West Latitudes stored within "$coord[1]")
    a. ($sort based on "latitude" – i.e., "$address.coord.1")

```
> db.restaurants.aggregate([
... { $match : {cuisine : "American" } },
... { $group : {
...      "_id" : "$name" } }
... ,
... { $sort : {"address.coord.1" : -1} },
... { $limit : 3 }
... ] )
{ "_id" : "Land & Sea Restaurant" }
{ "_id" : "The Dugout" }
{ "_id" : "Court Square Diner" }
>
```

5. Write a mongoDB query to find the three restaurants with the highest total score, that are not in Manhattan and do not serve French cuisine. (Hint : use $nor within the $match)

    a. (Utilizes a combination of many mongoDB features: .aggregate(), $match, $nor, $unwind, $group, $sum, $sort, $limit)

```
> db.restaurants.aggregate([
... { $match : {
...     $nor : [
...       {cuisine : "French"},
...       {borough : "Manhattan"}
...     ] } },
... { $unwind : "$grades" },
... { $group : {
...     "_id" : "$name",
...     "total_score" : {$sum : "$grades.score" } } },
... { $sort : {"total_score" : -1} },
... { $limit : 3 }
... ] )
{ "_id" : "Subway", "total_score" : 7746 }
{ "_id" : "Mcdonald'S", "total_score" : 6081 }
{ "_id" : "Dunkin' Donuts", "total_score" : 5351 }
```