1. From the command line run:
wget https://raw.githubusercontent.com/mongodb/docs-assets/primer-dataset/primer-dataset.json
 or manually download the above file

2.

Open a terminal and type: (server)
mongod

Open another terminal and type:
Run the following command in the shell:
mongoimport --db test --collection restaurants --drop --file primer-dataset.json

Open another terminal and type: (client)
mongo

***
db.restaurants.find({cuisine:"Italian"})
//display all attributes with grading = 1
db.restaurants.find({cuisine:"Italian"}, {grades:1})
//display all attributes except grade item
db.restaurants.find({cuisine:"Italian"}, {grades:0})

3. Find out how many documents are there in the restaurants collection:
   db.restaurants.count()

4. Display the first restaurant in the collection:
   db.restaurants.find().limit(1)

5. Find all the restaurants where the cuisine is Italian
   db.restaurants.find({cuisine:"Italian"})

5. Find the restaurants that satisfy the following criteria:
It should be in Manhattan burrough and the cuisine should be Italian
   db.restaurants.find({cuisine:"Italian",borough:"Manhattan"})
   ** below count nums

db.restaurants.find({cuisine:"Italian",borough:"Manhattan"}).count()

Also, find the count of such restaurants:

6. Display just the names of those restaurants where the cuisine is Indian and who have received the grade A at least once.

```
db.restaurants.find({cuisine:"Indian", "grades.grade":"A"}).count()
```

7. How many different types of cuisines are there in the collection, also display their names

```
db.restaurants.distinct("cuisine")
db.restaurants.distinct("cuisine”).length
```

8. Find the count of restaurants grouped by borough:

```
**
db.restaurants.aggregate([
    {$group: {"_id":"cuisine", "count": {$sum: 1}}}
    ])
```

9. Find the count of Chinese restaurants grouped by zipcode

```
db.restaurants.find({cuisine:"Chinese"}).aggregate(
    [
        {$group: {"_id":"$cuisine", "count": {$sum:1}}}
    ]
);
**

db.restaurants.aggregate(
    [
        {$match:{"cuisine”:”Chinese"}},
        {$group: {"_id":"$cuisine", "count": {$sum:1}}}
    ]
);
```

10. Find the count of Pizza restaurants in Brooklyn grouped by zipcode

```
    db.restaurants.aggregate(
    [
        {$match:{"cuisine”:”Chinese"}},
        {$group: {"_id":"$borough”, "count": {$sum:1}}}
    ]
);
```

11. Insert the orders records and find the todal amount grouped by customer id.

```
db.orders.insert({cust_id: "A123", amount: 500, status: "A"})
db.orders.insert({cust_id: "A123", amount: 250, status: "A"})
db.orders.insert({cust_id: "B212", amount: 200, status: "A"})
db.orders.insert({cust_id: "A123", amount: 300, status: "D"})
```

```
db.orders.mapReduce(
                                                    function()
{ emit(this.cust_id, this.amount); },
                                                    function(key, values)
{ return Array.sum(values)},
                                                    {       query: {status:
"A"},
                                                            out: "order_totals"
                                                    }
                                                    ).find()
```

12. Run mapreduce on the restaurants dataset to return number of restaurants grouped by cuisine in the Manhattan borough