Automobile Backup Products Booking System

Milestone: Implementation in NoSQL

Venkata Krishnan
857-210-1207
ravichandran.ve@northeastern.edu

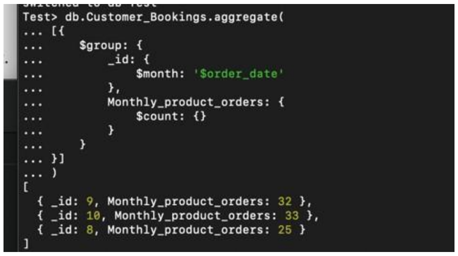
IMPLEMENTATION OF RELATIONAL MODEL IN NoSQL:

We have used MongoDB to perform the implementation of our relational database in NoSQL. We have added two tables namely Automobile_Spare_Products and Customer_Bookings to our MongoDB and loaded the data for both the tables by uploading their respective CSV files via MongoDB Compass.

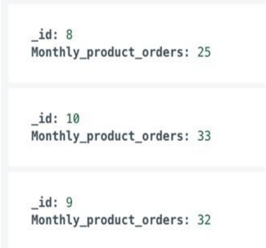
Now, we can execute a few queries to analyze the data present in the NoSQL database collections.

Query 1: Find the monthly sales of the automobile spare products.

```
db.Customer_Bookings.aggregate(
[{
    $group: {
        _id: {
            $month: '$order_date'
        },
        Monthly_product_orders: {
            $count: {}
        }
    }
}]
)
```



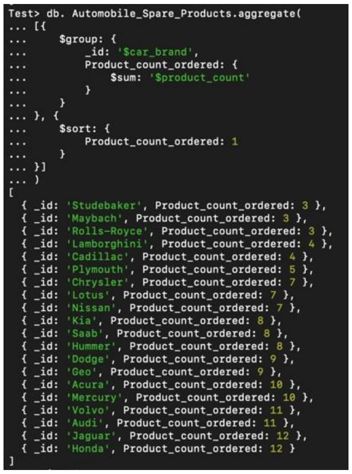
```
switched to db Test
Test> db.Customer_Bookings.aggregate(
... [{
...     $group: {
...         _id: {
...             $month: '$order_date'
...         },
...         Monthly_product_orders: {
...             $count: {}
...         }
...     }
... }]
... )
[
  { _id: 9, Monthly_product_orders: 32 },
  { _id: 10, Monthly_product_orders: 33 },
  { _id: 8, Monthly_product_orders: 25 }
]
```



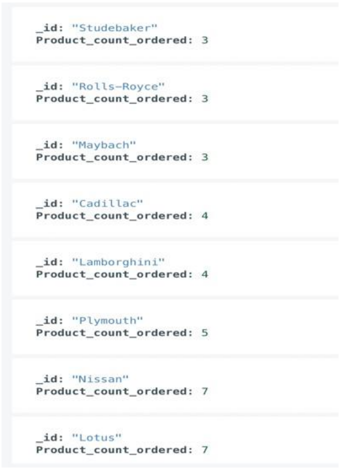
```
_id: 8
Monthly_product_orders: 25

_id: 10
Monthly_product_orders: 33

_id: 9
Monthly_product_orders: 32
```

Query 2: Find the automobile spare products sales of each Car Brands.

```
db. Automobile_Spare_Products.aggregate(
[{
    $group: {

        _id: '$car_brand',
        Product_count_ordered: {
            $sum: '$product_count'
        }
    }
}, {
    $sort: {
        Product_count_ordered: 1
    }
}]
)
```



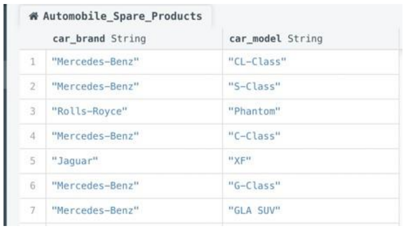
```
Test> db. Automobile_Spare_Products.aggregate(
... [{
...     $group: {
...         _id: '$car_brand',
...         Product_count_ordered: {
...             $sum: '$product_count'
...         }
...     }
... }, {
...     $sort: {
...         Product_count_ordered: 1
...     }
... }]
... )
[
  { _id: 'Studebaker', Product_count_ordered: 3 },
  { _id: 'Maybach', Product_count_ordered: 3 },
  { _id: 'Rolls-Royce', Product_count_ordered: 3 },
  { _id: 'Lamborghini', Product_count_ordered: 4 },
  { _id: 'Cadillac', Product_count_ordered: 4 },
  { _id: 'Plymouth', Product_count_ordered: 5 },
  { _id: 'Chrysler', Product_count_ordered: 7 },
  { _id: 'Lotus', Product_count_ordered: 7 },
  { _id: 'Nissan', Product_count_ordered: 7 },
  { _id: 'Kia', Product_count_ordered: 8 },
  { _id: 'Saab', Product_count_ordered: 8 },
  { _id: 'Hummer', Product_count_ordered: 8 },
  { _id: 'Dodge', Product_count_ordered: 9 },
  { _id: 'Geo', Product_count_ordered: 9 },
  { _id: 'Acura', Product_count_ordered: 10 },
  { _id: 'Mercury', Product_count_ordered: 10 },
  { _id: 'Volvo', Product_count_ordered: 11 },
  { _id: 'Audi', Product_count_ordered: 11 },
  { _id: 'Jaguar', Product_count_ordered: 12 },
  { _id: 'Honda', Product_count_ordered: 12 }
]
```



```
_id: "Studebaker"
Product_count_ordered: 3

_id: "Rolls-Royce"
Product_count_ordered: 3

_id: "Maybach"
Product_count_ordered: 3

_id: "Cadillac"
Product_count_ordered: 4

_id: "Lamborghini"
Product_count_ordered: 4

_id: "Plymouth"
Product_count_ordered: 5

_id: "Nissan"
Product_count_ordered: 7

_id: "Lotus"
Product_count_ordered: 7
```

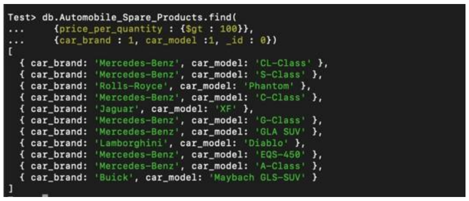
Query 3: Find Car Brands and Car Models with automobile spare products unit price greater than 100.

```
db.Automobile_Spare_Products.find(
        {price_per_quantity : {$gt : 100}},
        {car_brand : 1, car_model :1, _id : 0}
)
```



Automobile_Spare_Products

	car_brand String	car_model String
1	"Mercedes-Benz"	"CL-Class"
2	"Mercedes-Benz"	"S-Class"
3	"Rolls-Royce"	"Phantom"
4	"Mercedes-Benz"	"C-Class"
5	"Jaguar"	"XF"
6	"Mercedes-Benz"	"G-Class"
7	"Mercedes-Benz"	"GLA SUV"



```
Test> db.Automobile_Spare_Products.find(
...     {price_per_quantity : {$gt : 100}},
...     {car_brand : 1, car_model :1, _id : 0})
[
  { car_brand: 'Mercedes-Benz', car_model: 'CL-Class' },
  { car_brand: 'Mercedes-Benz', car_model: 'S-Class' },
  { car_brand: 'Rolls-Royce', car_model: 'Phantom' },
  { car_brand: 'Mercedes-Benz', car_model: 'C-Class' },
  { car_brand: 'Jaguar', car_model: 'XF' },
  { car_brand: 'Mercedes-Benz', car_model: 'G-Class' },
  { car_brand: 'Mercedes-Benz', car_model: 'GLA SUV' },
  { car_brand: 'Lamborghini', car_model: 'Diablo' },
  { car_brand: 'Mercedes-Benz', car_model: 'EQS-450' },
  { car_brand: 'Mercedes-Benz', car_model: 'A-Class' },
  { car_brand: 'Buick', car_model: 'Maybach GLS-SUV' }
]
```