PICT, PUNE

33271

①

Assignment No 2
ADBMS Lab

Roll No - 33271.

Problem statement - Implement Map-Reduce and aggregation indexing with suitable example in MONGODB. Demonstrate the following
i) Aggregation framework
ii) Create and drop different types of index
iii) explain() to show advantage of indices.

Objectives: 1) To understand the concept of map reduce in MongoDB
2) To implement the concept of document oriented DB

Theory:-

1) **Map Reduce** - It is a data processing paradigm for condensing large volume of data into useful aggregated results.

Syntax-

```
db.collectionName.mapReduce(
map → function() { emit (this.cust_id, this.amt); }
reduce → function(key, value) { return Array.sum(value) (code)
query → { Query: {} }
output → out: "output name"
    }
)
```

In the above example, MongoDB applies the map phase to each input document. The map function emits key-value pairs. For those keys that have multiple values, MongoDB applies the reduce phase which collects and condenses the aggregated data. MongoDB then stores the result in collection.

All Map-Reduce functions in MongoDB are Javascript.

2) Aggregation

The operations process data records and return computed results. Aggregation group values from multiple documents together and can perform a variety of operations on the grouped data to return single result.

The aggregation framework is modelled on the concept of data processing pipelines. Documents enter a multi-stage pipeline that transforms the document into an aggregated result.

Example with syntax:

```
db.orders.aggr Aggregate ([
        {$match : { status : "A" } }
        {$group : { _id : "$post_id", total : {$sum
                                        = "A"}
                                    }])
```

First stage:- The $match stage filters the document by the status field and passes to next stage.

PICT, PUNE

Second stage:- The $group stage filters documents by cust_id status field and calculate sum of amt for each unique cust_id.

3) Indexing:-

MongoDB uses indexing in order to make query processing more efficient without it. It must scan every document in mongodb and retrive only matched ones. Indices are special data structure that stores some info related to documents. It becomes easy for MongoDB to find documents.

Syntax:-

→ db.collection_name.createIndex({key:1})
'key' determies the field and 1 or -1 determines ascending or descending order.

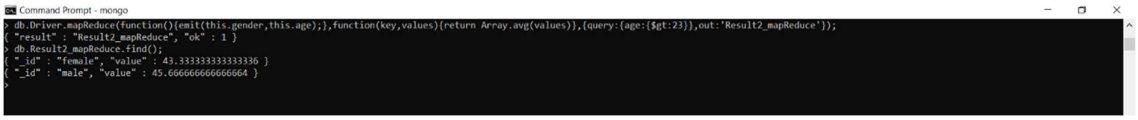
→ db.collection_name.dropIndex({key:1}) will drop the index

→ db.collection_name.getIndices() will retrive all desc of indices.

Screenshots of Implementation:

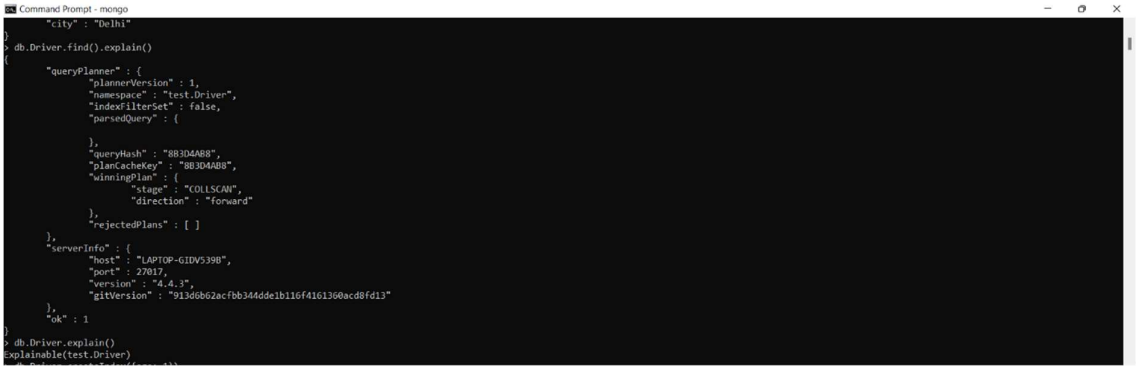
A) Map and Reduce Functions

1)Map and Reduce functions



2) Aggregate Function

3) Explain Keyword



B) Index

1) Create Index

2) Create Index with multiple attributes

3) Get Indexes

4) Drop Index

Conclusion- Understood map-reduce, aggregation framework and concept of Indexing in MongoDB.