DAYANAND SCIENCE COLLEGE, LATUR

DEPARTMENT OF COMPUTER SCIENCE

ASSIGNMANT SUBMISSION RECORD

ACADEMIC YEAR 2023-2024

ASSIGNMENT NO. 2 ROLL NO. 309

Name of Student: Kulkarni pranav sudhir

Class : Msc. CS-FY Subject : Computer science

Paper Title : Nosql with mongoDB Paper Code : SCMPSC-453

Date of Submission : 29-04-24

Student Signature :

Name of the Subject Teacher : Shazia mam.

Teacher Signature :

1] what is map reduce.

Ans:-

map reduce is a data processing programming model that helps to perform operations on large datasets and produce aggregated results. MongoDB provides the mapReduce() Function to perform the map-reduce operations. This Function has two main Functions, i.e:- map Function and reduce Function, The map Function is used to group all the mapped data. So, the data is independent mapped and reduced in different spaces and then combined together in the Function and the result will save to the specified genreally operated on large data sets only. using map reduce you can platform aggregation operations. such as max, avg on the data using come some key and it is similar to groupBy in sql. It performs on data independtly and parallel. let's try to understand the mapReduce() using the following example.

syntax:-

```
db.collectionName.mapReduce(
    map(),
    reduce(),
    query(),
   output {}
 );
```

- map() Function:-

it uses emit() Function in which it takes two parameters key and Value key. Here the key is on which we make groups like group by ages or names and the second parameter is on which aggregation is performed like avg(), sum (), is calculated on.

reduce () Function:-

It is the step in which we perform our aggregate Function like avg(), sum().

output():-

In this, we will specify the collection name where the result will be stored.

query:-

Here we will pass the query to Filter the result set.

2) what is partitioning and combining.

Ans:-

In the simplest Form, we think of a map-reduce job as having a single reduce Function. The output From all the map tasks running on the various nodes are concatenated together and sent into the reduce. while this will work, there are things we can do to increase the parallelism to reduce the data transfer.

The First thing we can do is increase parallelism by partitioning the output of the mappers. each reduce Function operators on the results of a single key This is a limitation it means you can't do anythings in the reduce that operators access keys- but it's also a benefit in that it allows you to run multiple reduce in parallel. To take advantage of this, the result of the mapper are divided up based the key on each processing node. Typically multiple keys are grouped together into partitions. the Framework then takes the data from all the nodes For one partition, combines it into a single groups For that partition, and sends it off to a reducer. multiple reducers can then operate on the partition. In parallel with the Final resuts merged together (This step is also called "shuffling" and the partitionsare sometimes referred to as "buckets" or "regions).

The next problems we can deal with is the amount of data being moved from node to node between the map and reduce stages. much of this data is repetitive, consisting of multiple key - value pairs for the map and reduce stages. the same key. A combinar functions cuts this data down by combining all the data for the same key into a single value. A combinar-function is, in essence, a reducer function - indeed, in many cases the same function can be used for combining as the final reducation. The reduce functions needs a special shape for this to work. Its output must match its input we all such a function a combinable reducer.

Not all reduce functions are combinable. consider a functions that counts the number of unique customers for a particular product. The map function for such on operation would need to emit. the product and the customer. - the reducer con then and count how many times each customer appears for a particular product.

Transactions:-

Transactions in the tranditional RDBMS sense, mean that you can start modifying the database with insert, update or delete commands over different tables and then decide if you want to keep the changes or not by using commit or rollback these Constructs are generally not available in NO sql solutions - a write either succeeds or fails. transactions at the single-document level are known as atomic transactions. transacations involving more than one operation are not possible, although there are - products such as RavenDB that do support transaction across multiple operations.

Availability :-

The CAP theoream dicates that we can have only two of Consistancy, Availability, and partition, tolerance. Document databases try to improve - on availability by replacing data using the master-slave setup. the same data is available on multiple nodes and the clients can get to the data even when the primary node is down. Usally, the application code does not have to determine if the primary node is available or not. MongoDB implement replication, providing high availability using replica sets.

4] what is a graph databases.

Ans:- we see a bunch of nodes related to each other. Nodes are entities that have properties, such as name. the node of martin is actually a node that has property of name set to martin.

we also see that edges have types, such as likes, author, and so on. these properties let us organize the nodes. ex:- the nodes martin and pramod have an edge Connecting them with a relationship type of friend. edges can have multiple properties. we can assign a property of since on the Friend relationship type between - martin and pramod. Relationship types have directional significance, the Friend relationship type is bidirectional but likes is not. when Down-likes.

NosqL Distilled, it does not automatically means - Nosql Distilled likes Down.

once we have a graph of these nodes and edges created, we can query the - graph in many ways, such as " get all nodes employed big Co that like NosqL Distilled:"

A query on the graph is also known as traversing the graph. An advantages of the graph databases is that we can changes the traversing requirements without having to changes the nodes or edges. If we want to "get all nodes that like NoSQL Distilled" we can do so without having to changes the existing data or the modal of the database, because we can traverse the graph any way we like.

In graph database, traversing the join or relationships is very Fast. the relationship between nodes is not calculated at query time but is actually persisted as a relationship. tranversing persisted relationships is Faster than calculating them For every query.