### 

**Experiment No.5**

**Title: NOSQL and MongoDB**

**Batch: B2** **Roll No.: 1914078** **Experiment No.:5**

**Aim: To implement a NOSQL database using MongoDB and execute queries.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Resources needed:MongoDB, Windows**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Theory**

MongoDB is an open-source document database and leading NoSQL database. MongoDB is written in C++.

MongoDB is a cross-platform, document oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document.

Database

Database is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases.

Collection

Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose.

Document

A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

The following table shows the relationship of RDBMS terminology with MongoDB.

|  |  |
| --- | --- |
| **RDBMS** | **MongoDB** |
| Database | Database |
| Table | Collection |
| Tuple/Row | Document |
| column | Field |
| Table Join | Embedded Documents |
| Primary Key | Primary Key (Default key \_id provided by mongodb itself) |

## Sample Document

Following example shows the document structure of a blog site, which is simply a comma separated key value pair.

{

\_id: ObjectId(7df78ad8902c)

title: 'MongoDB Overview',

description: 'MongoDB is no sql database',

by: 'MongoDb info’,

url: 'http://www.MongoDb.com',

tags: ['mongodb', 'database', 'NoSQL'],

likes: 100,

comments: [

{

user:'user1',

message: 'My first comment',

dateCreated: new Date(2011,1,20,2,15),

like: 0

},

{

user:'user2',

message: 'My second comments',

dateCreated: new Date(2011,1,25,7,45),

like: 5

}

]

}

**\_id** is a 12 bytes hexadecimal number which assures the uniqueness of every document. You can provide \_id while inserting the document. If you don’t provide then MongoDB provides a unique id for every document. These 12 bytes first 4 bytes for the current timestamp, next 3 bytes for machine id, next 2 bytes for process id of MongoDB server and remaining 3 bytes are simple incremental VALUE.

Data in MongoDB has a flexible schema.documents in the same collection. They do not need to have the same set of fields or structure, and common fields in a collection’s documents may hold different types of data.

## Some considerations while designing Schema in MongoDB

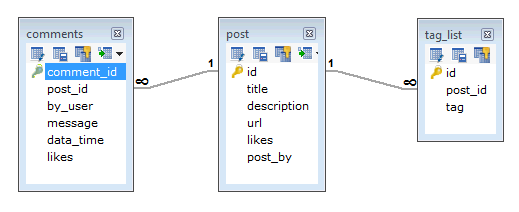
* Design your schema according to user requirements.
* Combine objects into one document if you will use them together. Otherwise separate them (but make sure there should not be need of joins).
* Duplicate the data (but limited) because disk space is cheap as compare to compute time.
* Do joins while write, not on read.
* Optimize your schema for most frequent use cases.
* Do complex aggregation in the schema.

## Example

Suppose a client needs a database design for his blog/website and see the differences between RDBMS and MongoDB schema design. Website has the following requirements.

* Every post has the unique title, description and url.
* Every post can have one or more tags.
* Every post has the name of its publisher and total number of likes.
* Every post has comments given by users along with their name, message, data-time and likes.
* On each post, there can be zero or more comments.

In RDBMS schema, design for above requirements will have minimum three tables.



While in MongoDB schema, design will have one collection post and the following structure −

{

\_id: POST\_ID

title: TITLE\_OF\_POST,

description: POST\_DESCRIPTION,

by: POST\_BY,

url: URL\_OF\_POST,

tags: [TAG1, TAG2, TAG3],

likes: TOTAL\_LIKES,

comments: [

{

user:'COMMENT\_BY',

message: TEXT,

dateCreated: DATE\_TIME,

like: LIKES

},

{

user:'COMMENT\_BY',

message: TEXT,

dateCreated: DATE\_TIME,

like: LIKES

}

]

}

## Why Use MongoDB?

* **Document Oriented Storage** − Data is stored in the form of JSON style documents.
* Index on any attribute
* Replication and high availability
* Auto-sharding
* Rich queries
* Fast in-place updates
* Professional support by MongoDB

## Where to Use MongoDB?

* Big Data
* Content Management and Delivery
* Mobile and Social Infrastructure
* User Data Management
* Data Hub

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Procedure:**

1. Installation of mongodb on windows

1. Create nosql database using mongodb
2. Insert data into collection

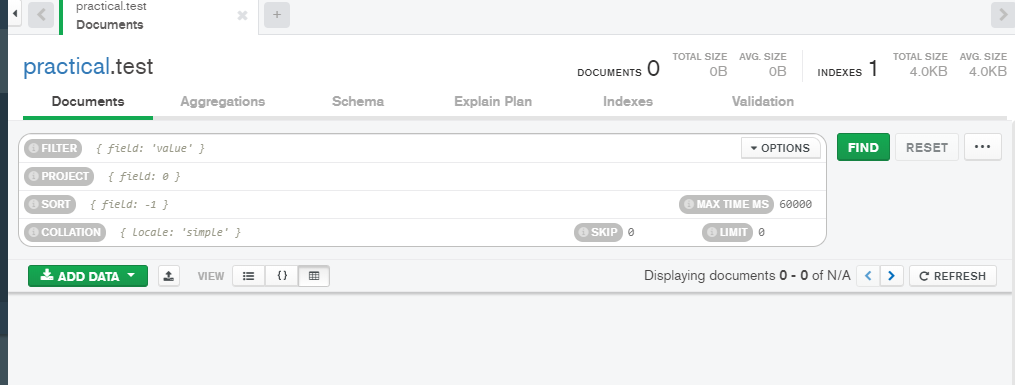
4. Update, Retrieve and delete entries from collection

5. Explore any 5 methods(not mentioned in the document)and use it on your database.

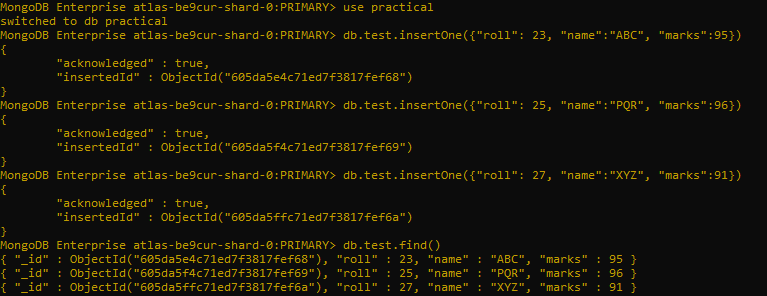
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Results: (Program printout with output)**

Created a database named “practical” and created a connection named “test” within it.



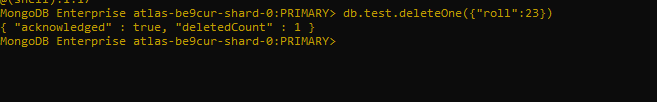
Insertion



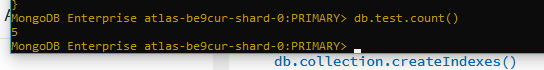
Updation



Deletion



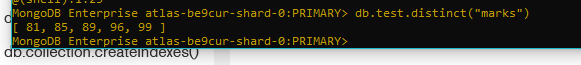
Count



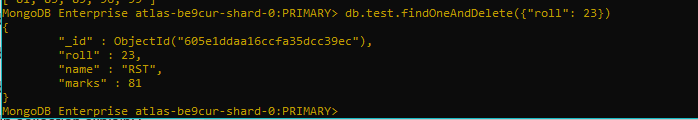
Data size



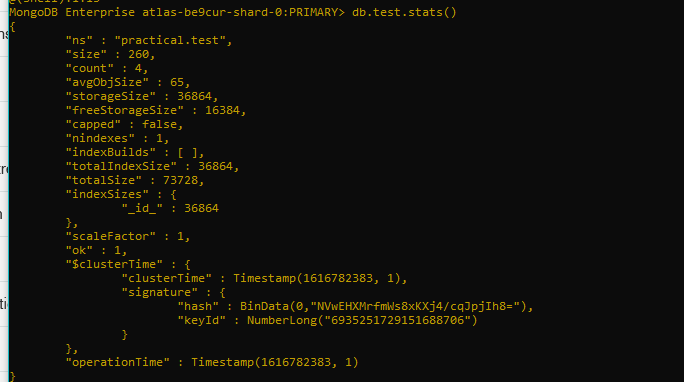
Distinct



Findone and delete



Stats



**Questions:**

**1. Explain the 5 methods used for these queries in detail.**

1. Distinct()- Finds the distinct values for a specified field across a single collection. [distinct](https://docs.mongodb.com/manual/reference/command/distinct/#dbcmd.distinct) returns a document that contains an array of the distinct values. The return document also contains an embedded document with query statistics and the query plan.
2. dataSize() - The [dataSize](https://docs.mongodb.com/manual/reference/command/dataSize/" \l "dbcmd.dataSize" \o "dataSize) command returns the data size for a set of data within a certain range. The amount of time required to return [dataSize](https://docs.mongodb.com/manual/reference/command/dataSize/" \l "dbcmd.dataSize" \o "dataSize) depends on the amount of data in the collection.
3. count() - Counts the number of documents in a collection or a view. Returns a value that contains this.
4. Stats(): Returns statistics that reflect the use state of a single database. The db.stats() method is a wrapper around the dbStats database command
5. findAndOneDelete():Deletes a single document based on the filter and sort criteria, returning the deleted document.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Outcomes:**

Design advanced database systems using Object Relational, Spatial and NOSQL Databases and its implementation

**Conclusion: (Conclusion to be based on outcomes achieved)**

We can conclude that we were successfully able to execute the install MongoDB, create a database and collection and execute the queries and commands.

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of faculty in-charge with date**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**References:**

1. Elmasri and Navathe, “Fundamentals of Database Systems”, Pearson Education
2. Raghu Ramakrishnan and Johannes Gehrke, “Database Management Systems” 3rd

Edition, McGraw Hill,2002

1. Korth, Silberchatz, Sudarshan, “Database System Concepts” McGraw Hill
2. http://www.bostongis.com/PrinterFriendly.aspx?content\_name=postgis\_tut01