Discussion 4.1 – What Is MongoDB?

Shane Hingtgen

4/3/23

Bellevue University

MongoDB is a very strong and adaptable database usable for a ton of applications. It is considered a document-oriented database and not a relational database (Bradshaw & Chodorow, 2019). By not making it a relational database it allows scaling to be easier (Bradshaw & Chodorow, 2019). With MongoDB you can create, read, update or delete data but it can also index, aggregation, special collection and index types, and file storage (Bradshaw & Chodorow, 2019).

According to our book MongoDB, the Definitive Guide, a document is defined as “the basic unit of data for mongoDB and is roughly equivalent to a row in a relation database management system.” (Bradshaw & Chodorow, 2019) Which is key-value pairs. And a collection can be described as “a table with a dynamic schema” (Bradshaw & Chodorow, 2019). Another way to look at a collection is that it’s just a group of documents.

In MongoDB there is no default way to auto-increment like the way SQL does. (*MongoDB - Auto-Increment Sequence*, n.d.) So we would have to built one. If we want the \_id to increase to stay unique we would have to use the follow code for example.

{

"\_id":1,

"plant\_name": "Indian Grass",

"category": "native Iowa prairie grasses"

}

Then according to tutorials point we will need to track the value for all the fields with

>db.createCollection("counters")

Next we add the document in the collection we just made

> db.counters.insert({

"\_id":"productid",

"sequence\_value": 0

})

WriteResult({ "nInserted" : 1 })

>

Then add this code so we can insert the sequence document in the collection

>db.counters.insert({\_id:"productid",sequence\_value:0})

Create the JavaScript function

>function getNextSequenceValue(sequenceName){

var sequenceDocument = db.counters.findAndModify({

query:{\_id: sequenceName },

update: {$inc:{sequence\_value:1}},

new:true

});

return sequenceDocument.sequence\_value;

}

The main differences between MySQL and MongoDB are as followed.

* MongoDB is a non-relational database where as MySQL is a Relational Database.
* MySQL stores the data with tables and rows
* MongoDB uses documents like JSON, BSON to store the data
* MySQL schemas will need to be defined before creating the data and tables, and the data must match the schemas
* MongoDB has documents that have key-value pairs

References:

MongoDB. (n.d.). *Comparing The Differences - MongoDB Vs MySQL*. Retrieved April 3, 2023, from https://www.mongodb.com/compare/mongodb-mysql

Bradshaw, S., & Chodorow, K. (2019). *MongoDB: Powerful and Scalable Data Storage*. O’Reilly Media.

*MongoDB - Auto-Increment Sequence*. (n.d.). Tutorials Point. Retrieved April 3, 2023, from https://www.tutorialspoint.com/mongodb/mongodb\_autoincrement\_sequence.htm