**MONGODB**

* CRUD Operations (Commands)
* mongo -> connect to mongodb.

//CREATE =>

* use database -> create (if doesn’t exist) and use ‘database’ database.
* db.createCollection(“users”) -> create a “users” collection.

//INSERT=>

* db.users.insertOne({name: "Avantika"}) -> insert one data into collection “users”
* db.users.insert({ ->Insert data into collection “users”

title : “Tutorial”,

author : “Avantika Khanna”,

content : “Database”

})

// FIND =>

* db.coll.findOne({title: “MongoBD” }) -> returns a single document.
* db.users.find() -> find all documents in “users” collection.
* db.users.findOneAndUpdate( -> find and modify document

{ titie : “monodb” },

{ $set : {content : “ tutorial is here” }}

)

* db.users.findOneAndDelete( ->find one and delete in document

{ author : “John” }

)

* db.users.findOneAndReplace( ->find one and replace in document

{ title : “monodb”,

{ title : “mongodb tutorial” ,content : “Guide to mongodb” }

)

// UPDATE=>

* db.users.updateOne( -> Update the “users” collection

{ name : “Avantika “ },

{ $set : { age : 20 } }

)

* db.users.updateOne(

{ name : “Avantika”},

{ $unset : {“year” : 2014 } }

)

// DELETE=>

* db.users.deleteOne({ name : “Aditya” }) -> delete a documents from “users” collection.
* db.users.deleteMany({ author: “John”}) -> delete multiple documents from “users” collection.
* db.coll.findOneAndDelete({"name": "Max"}) ->FIND AND DELETE .

//DROP =>

* db.users.drop() -> drop the entire “users” collection.
* Db.dropdatabase() -> drop the database itself.
* **MONGODB INDEXING**

Indexing enhances query performance and allows for efficient data retrieval in MongoDB.

* db.users.createIndex({username : 1})

Create a field index in “username” field in “users” collection.

* db.users.createIndex({username : 1, content : 1})

Create a compound index in “title” amd “content” in “users” collection.

* db.users.createIndex({content : “text” })

Create a text index in “content ” field in “users” collection.

* db.users.createIndex({email : 1}, {unique : true} )

Create a unique index in “email” field in “users” collection.

* db.users.getIndexes() -> get the list of indexes on “users” collection.
* db.users.dropIndex(“username\_1”) -> drop the index in “username” field in ‘user’ collection.
* **DATA MODELING IN MONGODB**

Data modeling in MongoDB involves designing schemas and relationships between documents.

Key Concepts:

1.)Documents: The basic unit of data in MongoDB, represented as BSON (Binary JSON) objects.

2.)Collections: Groups of documents, similar to tables in relational databases.

3.)Embedded Documents: Documents can contain sub-documents, allowing nested data structures.

4.)References: Documents can reference other documents, enabling relationships between data.