# FULL STACK WITH MERN (JAVA) ASSIGNMENT-4

Name: Annavarapu Radhika

Roll No:20NN1A0504

College Name: Vignan’s Nirula Institute of Technology and Science For Women (VNITSW)

Year:4th B-Tech (CSE)

# ASSIGNMENT-4: Creating a Database Using MongoDB and Mongosh

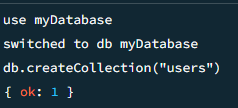
## OBJECTIVE:

The objective of this assignment is to familiarize yourself with MongoDB and its command-line interface, Mongosh, and to understand how to create, manage,and query databases and collections in MongoDB.

## PROCESS

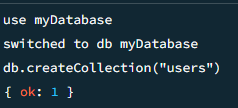
1. **DATABASE SETUP**: Create a new MongoDB database called myDatabase.
   * use myDatabase

### Output:



1. **COLLECTION CREATION:** Create a collection named users within the myDatabase database.
   * db.createCollection("users")

### Output:



1. **DOCUMENT INSERTION:** Insert at least three documents into the users collection, each representing a user with fields such as name, email, and age.

* db.users.insertMany([

{ name: "John Doe", email: "[john@example.com"](mailto:john@example.com), age: 25 },

{ name: "Sai", email: "[sai@example.com"](mailto:sai@example.com), age: 35 },

{ name: "Alice Johnson", email: "[alice@example.com"](mailto:alice@example.com), age: 30

},

{ name: "pavan", email: "[pavan@example.com"](mailto:pavan@example.com), age: 19 },

{ name: "lohi", email: "[lohi@example.com"](mailto:lohi@example.com), age: 20 },

{ name: "shubman gill", email: "[shubman@example.com"](mailto:shubman@example.com), age: 24 },

])

### Output:



1. **QUERYING:** Write queries to retrieve: All users from the users collection.

* db.users.find()

### Output:



1. **RETRIVING USERS WHOSE AGE IS GREATER THAN 30**: Users with an age greater than or equal to 30.
   * db.users.find({ age: { $gte: 30 } })

### Output:



1. **UPDATE OPERATION:** Update the age of a user with a specific email address.



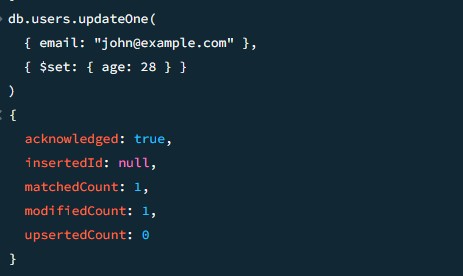
db.users.updateOne(

{ email: "[john@example.com"](mailto:john@example.com) },

{ $set: { age: 28 } }

)

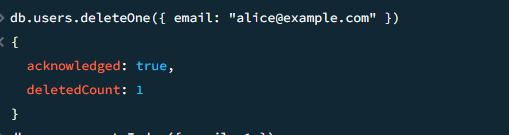
### Output:



1. **DELETION OPERATION:** Delete a user document based on a specific email address.

* db.users.deleteOne({ email: "[alice@example.com"](mailto:alice@example.com) })

### Output:



1. **INDEX CREATION:** Create an index on the email field of the users collection.
   * db.users.createIndex({ email: 1 }, { unique: true })

### Output:

