Creating a Database Using MongoDB and Mongosh

Database Setup and Collection Creation,Document Insertion ,Update Operation,deletion,index

```
// Require the MongoDB driver
const MongoClient = require('mongodb').MongoClient;

// Connection URI
const uri = "mongodb://localhost:27017";

// Database Name
const dbName = "myDatabase";

// Create a new MongoClient
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: true

// Connect to the MongoDB server
client.connect(function(err) {
  if(err) {
    console.error("Error connecting to MongoDB:", err);
    return;
  }

  console.log("Connected successfully to server");

  const db = client.db(dbName);

  // Create a collection
  db.createCollection("myCollection", function(err, res) {
    if(err) {
      console.error("Error creating collection:", err);
      return;
    }
    console.log("Collection created!");

    // Close the connection
    client.close();
  });
});
```

Document Insertion

javascript Copy code

```javascript
// Require the MongoDB driver
const MongoClient = require('mongodb').MongoClient;

// Connection URI
const uri = "mongodb://localhost:27017";

// Database Name
const dbName = "myDatabase";

// Sample documents to insert
const documents = [
  { name: "John", age: 30, city: "New York" },
  { name: "Alice", age: 25, city: "London" },
  { name: "Bob", age: 35, city: "Paris" }
];

// Create a new MongoClient
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: true

// Connect to the MongoDB server
client.connect(function(err) {
  if(err) {
    console.error("Error connecting to MongoDB:", err);
    return;
  }

  console.log("Connected successfully to server");

  const db = client.db(dbName);

  // Get the collection
  const collection = db.collection("myCollection");

  // Insert documents
  collection.insertMany(documents, function(err, result) {
    if(err) {
      console.error("Error inserting documents:", err);
      return;
    }

    console.log(`${result.insertedCount} documents inserted`);

    // Close the connection
    client.close();
  });
});
```

javascript Copy code

```javascript
// Require the MongoDB driver
const MongoClient = require('mongodb').MongoClient;

// Connection URI
const uri = "mongodb://localhost:27017";

// Database Name
const dbName = "myDatabase";

// Sample documents to insert
const documents = [
  { name: "John", age: 30, city: "New York" },
  { name: "Alice", age: 25, city: "London" },
  { name: "Bob", age: 35, city: "Paris" }
];

// Create a new MongoClient
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: true

// Connect to the MongoDB server
client.connect(function(err) {
  if(err) {
    console.error("Error connecting to MongoDB:", err);
    return;
  }

  console.log("Connected successfully to server");

  const db = client.db(dbName);

  // Get the collection
  const collection = db.collection("myCollection");

  // Insert documents
  collection.insertMany(documents, function(err, result) {
    if(err) {
      console.error("Error inserting documents:", err);
      return;
    }

    console.log(`${result.insertedCount} documents inserted`);

    // Close the connection
    client.close();
  });
});
```

```
// Require the MongoDB driver
const MongoClient = require('mongodb').MongoClient;

// Connection URI
const uri = "mongodb://localhost:27017";

// Database Name
const dbName = "myDatabase";

// Create a new MongoClient
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: tru

// Connect to the MongoDB server
client.connect(function(err) {
  if(err) {
    console.error("Error connecting to MongoDB:", err);
    return;
  }

  console.log("Connected successfully to server");

  const db = client.db(dbName);

  // Get the collection
  const collection = db.collection("myCollection");

  // Update a document
  const filter = { name: "John" }; // Filter to match documents to update
  const update = { $set: { age: 31, city: "Los Angeles" } }; // Update operation
  collection.updateOne(filter, update, function(err, result) {
    if(err) {
      console.error("Error updating document:", err);
      return;
    }

    console.log(`${result.modifiedCount} document updated`);

    // Close the connection
    client.close();
  });
});
```

javascript Copy code

```javascript
// Require the MongoDB driver
const MongoClient = require('mongodb').MongoClient;

// Connection URI
const uri = "mongodb://localhost:27017";

// Database Name
const dbName = "myDatabase";

// Create a new MongoClient
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: tru

// Connect to the MongoDB server
client.connect(function(err) {
  if(err) {
    console.error("Error connecting to MongoDB:", err);
    return;
  }

  console.log("Connected successfully to server");

  const db = client.db(dbName);

  // Get the collection
  const collection = db.collection("myCollection");

  // Delete a document
  const filter = { name: "John" }; // Filter to match documents to delete
  collection.deleteOne(filter, function(err, result) {
    if(err) {
      console.error("Error deleting document:", err);
      return;
    }

    console.log(`${result.deletedCount} document deleted`);

    // Close the connection
    client.close();
  });
});
```

javascript Copy code

```javascript
// Require the MongoDB driver
const MongoClient = require('mongodb').MongoClient;

// Connection URI
const uri = "mongodb://localhost:27017";

// Database Name
const dbName = "myDatabase";

// Create a new MongoClient
const client = new MongoClient(uri, { useNewUrlParser: true, useUnifiedTopology: tru

// Connect to the MongoDB server
client.connect(function(err) {
  if(err) {
    console.error("Error connecting to MongoDB:", err);
    return;
  }

  console.log("Connected successfully to server");

  const db = client.db(dbName);

  // Get the collection
  const collection = db.collection("myCollection");

  // Delete a document
  const filter = { name: "John" }; // Filter to match documents to delete
  collection.deleteOne(filter, function(err, result) {
    if(err) {
      console.error("Error deleting document:", err);
      return;
    }

    console.log(`${result.deletedCount} document deleted`);

    // Close the connection
    client.close();
  });
});
```