knowledgehut
upGrad

Introduction to NoSQL - MongoDB

Pre-requisites

Hope you have gone through the self-learning content for this session on the PRISM portal.

By the End of this Session, You will be Able to:

- Learn the basics of NoSQL databases.
- Understand the various types of NoSQL databases.
- Understand how Document Databases to store data in JSON files.
- Learn the basics of MongoDB, which is a document database.

What’s in It for Me?

Poll Time

Q. What does NoSQL stand for?

a. No Signature Query Language

b. Non-SQL Database

c. New Object Query Language

d. Notation of Structured Query Language

Poll Time

Q. What does NoSQL stand for?

a. No Signature Query Language

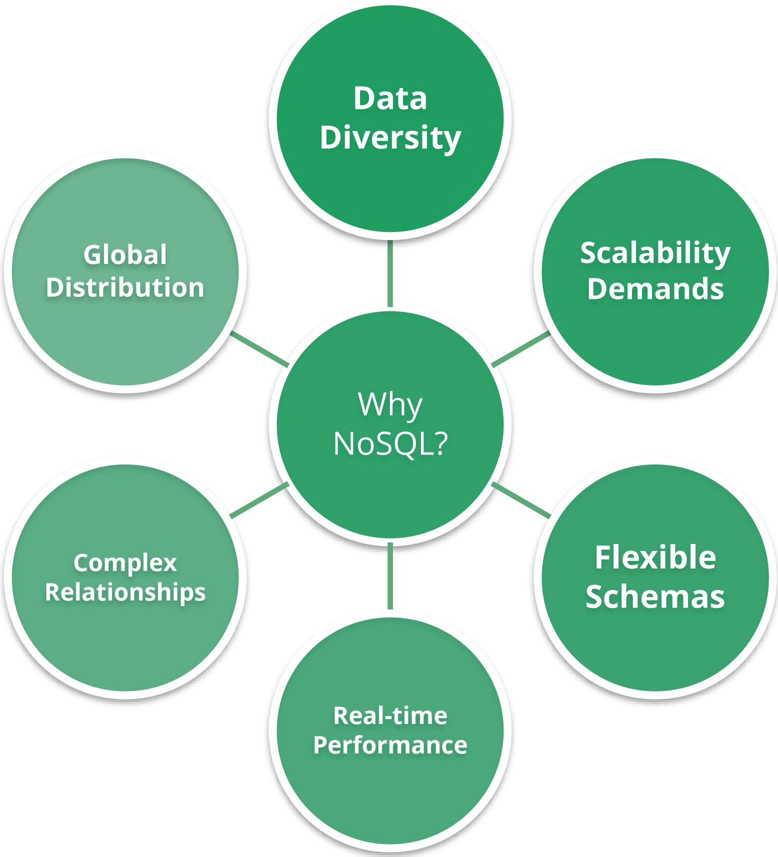
▌ **b. Non-SQL Database**

c. New Object Query Language

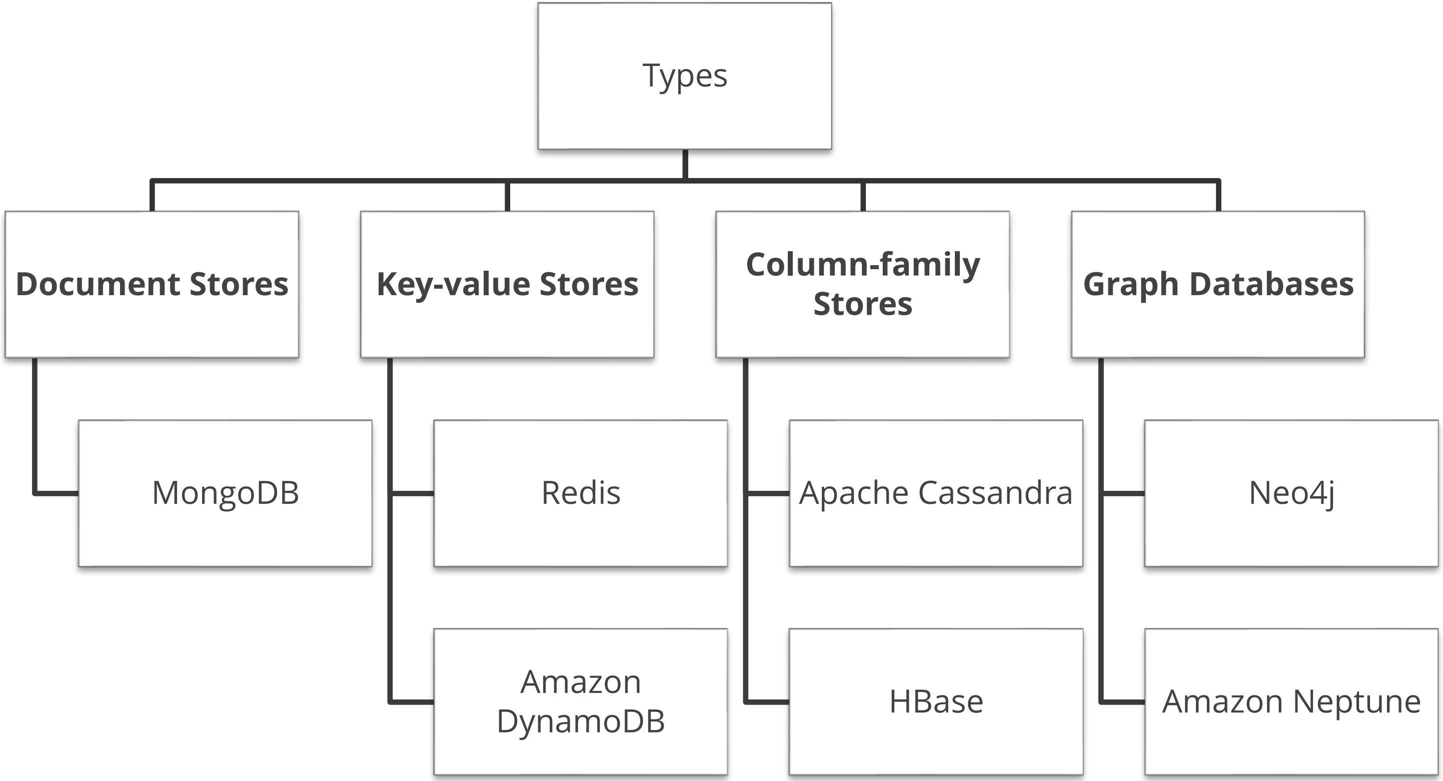
d. Notation of Structured Query Language

📄 NoSQL and Document Databases

Introduction to NoSQL Databases



Types of NoSQL Databases



Types

Document Stores

MongoDB

Key-value Stores

Redis

Amazon DynamoDB

Column-family Stores

Apache Cassandra

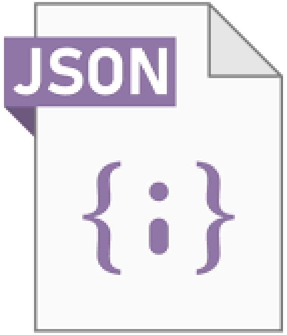
HBase

Graph Databases

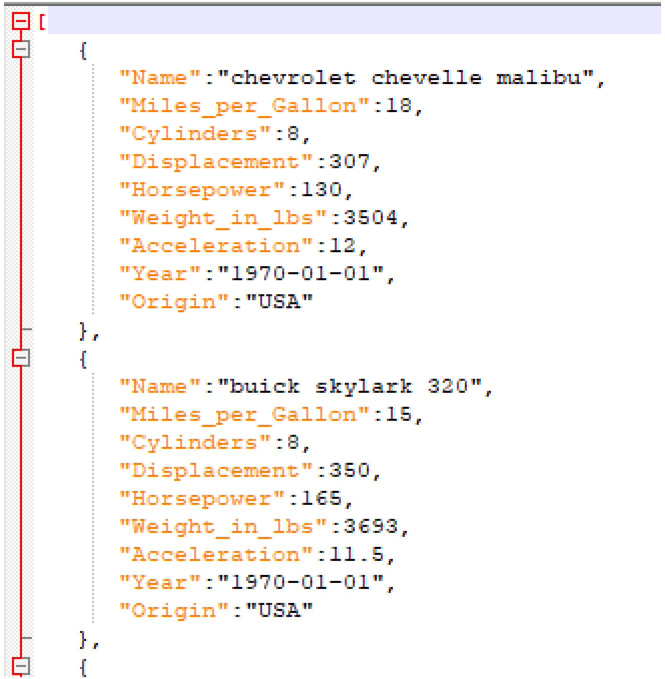
Neo4j

Amazon Neptune

Understanding Document Databases



Sample JSON File



```
[
    {
        "Name":"chevrolet chevelle malibu",
        "Miles_per_Gallon":18,
        "Cylinders":8,
        "Displacement":307,
        "Horsepower":130,
        "Weight_in_lbs":3504,
        "Acceleration":12,
        "Year":"1970-01-01",
        "Origin":"USA"
    },
    {
        "Name":"buick skylark 320",
        "Miles_per_Gallon":15,
        "Cylinders":8,
        "Displacement":350,
        "Horsepower":165,
        "Weight_in_lbs":3693,
        "Acceleration":11.5,
        "Year":"1970-01-01",
        "Origin":"USA"
    },
    {
```

Understanding Document Databases

- Part of NoSQL databases category.
- Designed for handling semi-structured and unstructured data.
- Store, retrieve, and manage data in document format.

Pop Quiz

Q. Which of the following is a characteristic of NoSQL databases?

a. Structured schema

b. ACID transactions

c. Fixed data types

d. Horizontal scalability

Pop Quiz

Q. Which of the following is a characteristic of NoSQL databases?

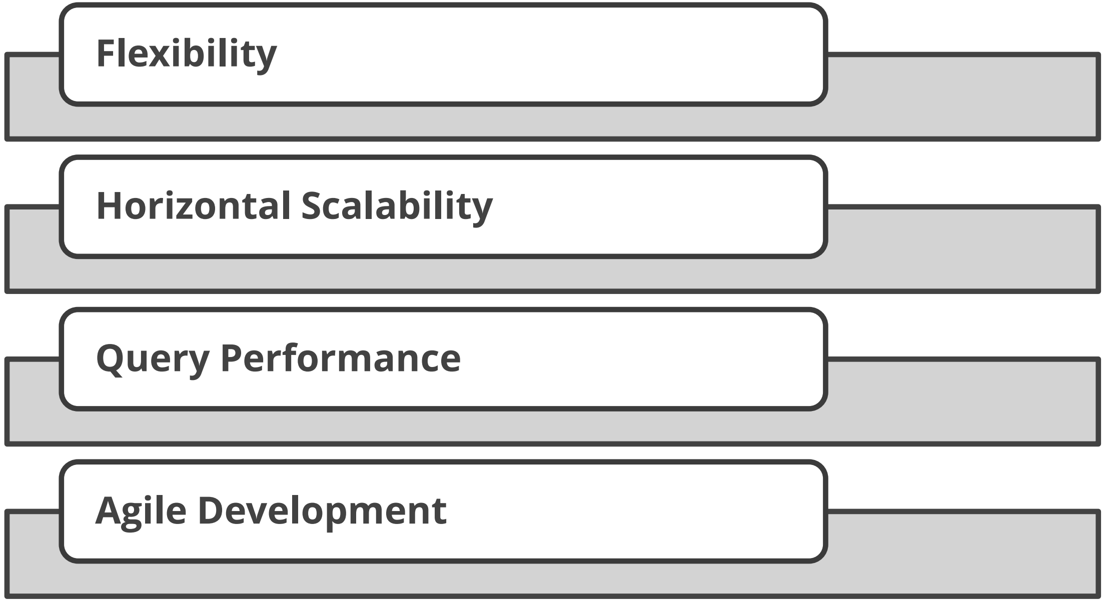
a. Structured schema

b. ACID transactions

c. Fixed data types

▌ **d. Horizontal scalability**

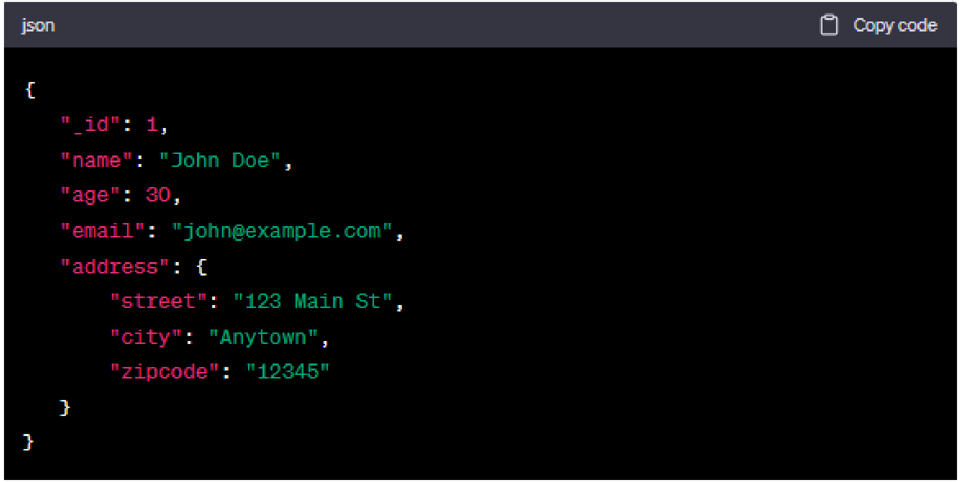
Benefits of Document Databases



- Flexibility
- Horizontal Scalability
- Query Performance
- Agile Development

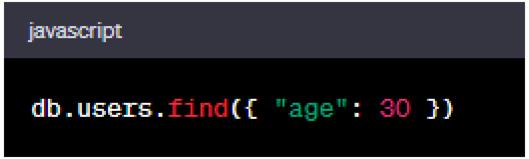
Querying in Document Databases

Querying: Extracting specific information from a database.



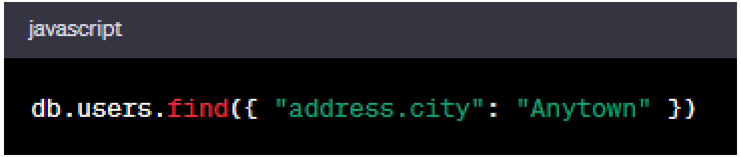
```json
{
   "_id": 1,
   "name": "John Doe",
   "age": 30,
   "email": "john@example.com",
   "address": {
       "street": "123 Main St",
       "city": "Anytown",
       "zipcode": "12345"
   }
}
```

Find Documents with Specific Field



```javascript
db.users.find({ "age": 30 })
```

Find Documents with Nested Field



```javascript
db.users.find({ "address.city": "Anytown" })
```

Real-world Applications of Document Databases

Content Management System

E-Commerce and Catalogs

Poll Time

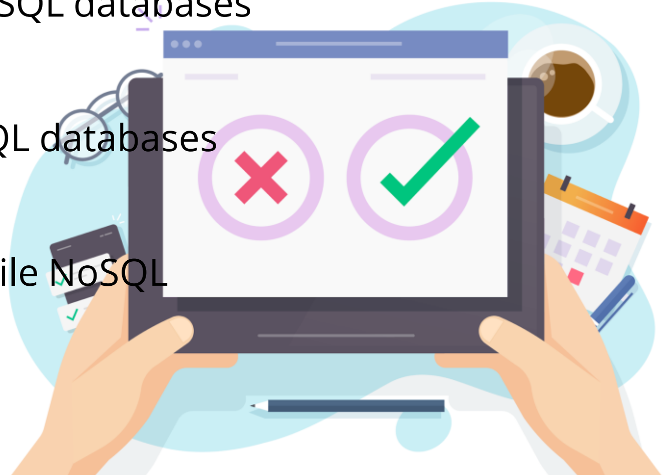
Q. Which of the following is a key difference between SQL and NoSQL databases?

a. SQL databases are schema-less, while NoSQL databases have a fixed schema

b. SQL databases primarily use JSON for data storage, while NoSQL databases use tabular structures

c. SQL databases guarantee strict ACID compliance, while NoSQL databases offer more flexible consistency models

d. SQL databases are better suited for horizontal scalability, while NoSQL databases are designed for vertical scalability



Poll Time

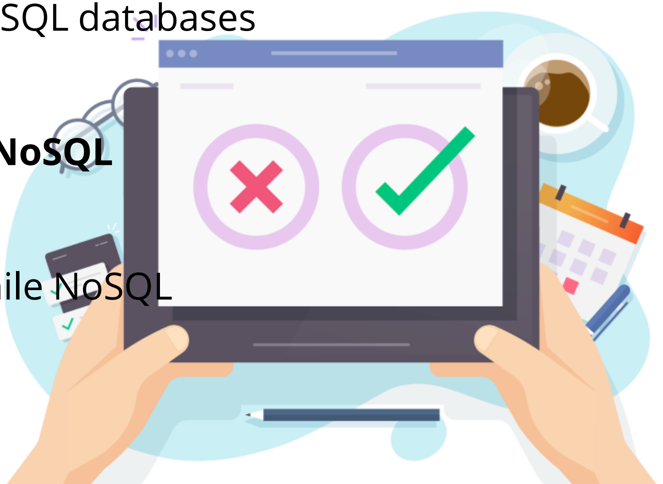
Q. Which of the following is a key difference between SQL and NoSQL databases?

a. SQL databases are schema-less, while NoSQL databases have a fixed schema

b. SQL databases primarily use JSON for data storage, while NoSQL databases use tabular structures

c. SQL databases guarantee strict ACID compliance, while NoSQL databases offer more flexible consistency models

d. SQL databases are better suited for horizontal scalability, while NoSQL databases are designed for vertical scalability



Q&A

MongoDB Basics

Introduction to MongoDB

- MongoDB is a leading NoSQL database.
- It is a part of the document database category.
- MongoDB is designed for flexible, scalable, and high-performance data storage and retrieval.

Architecture of MongoDB

Client

- Application interacting with MongoDB using drivers or libraries.

MongoDB Servers

- Store data and manage queries.

Replica Sets

- Multiple servers for redundancy and high availability.

Sharding

- Distributes data across multiple servers for horizontal scaling.

MongoDB Terminology

Document

- A fundamental unit of data storage in MongoDB.

Collection

- A grouping of MongoDB documents.

Database

- A container for collections.

Field

- A key-value pair within a MongoDB document.

Query

- A request to retrieve data from a MongoDB collection.

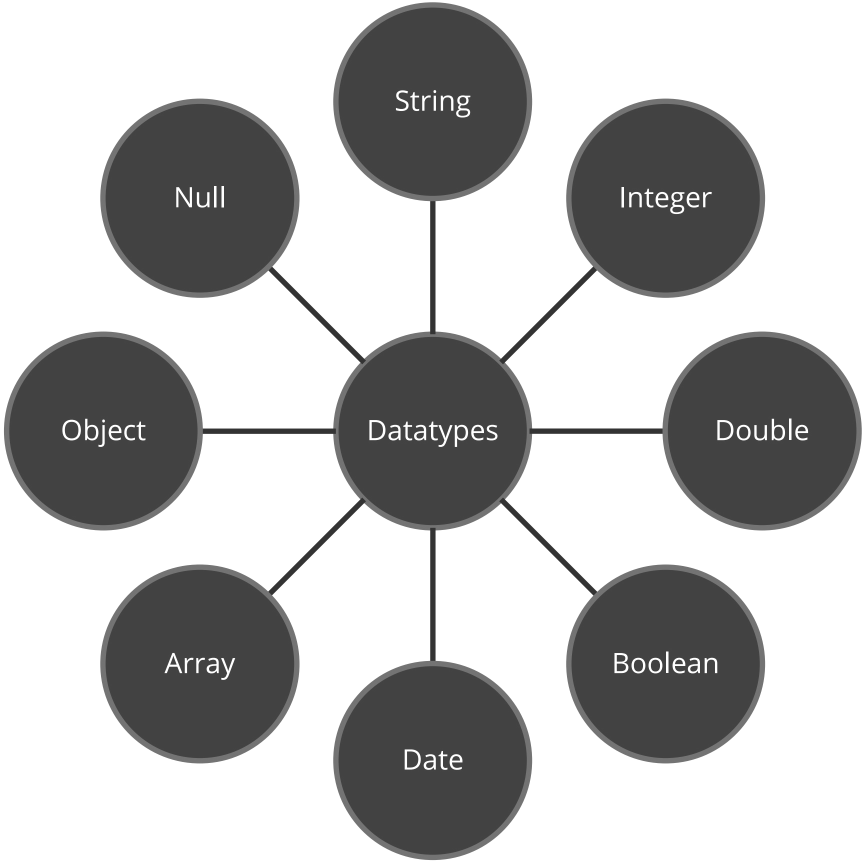
Aggregation

- A framework for performing advanced data transformations and processing.

Atlas

- MongoDB's managed cloud-based service.

MongoDB Data Types



Pop Quiz

Q. What is the primary purpose of sharding in MongoDB architecture?

a. Ensuring data consistency in a single server

b. Storing data in a denormalized format

c. Distributing data across multiple servers for horizontal scalability

d. Managing data replication for high availability

Pop Quiz

Q. What is the primary purpose of sharding in MongoDB architecture?

a. Ensuring data consistency in a single server

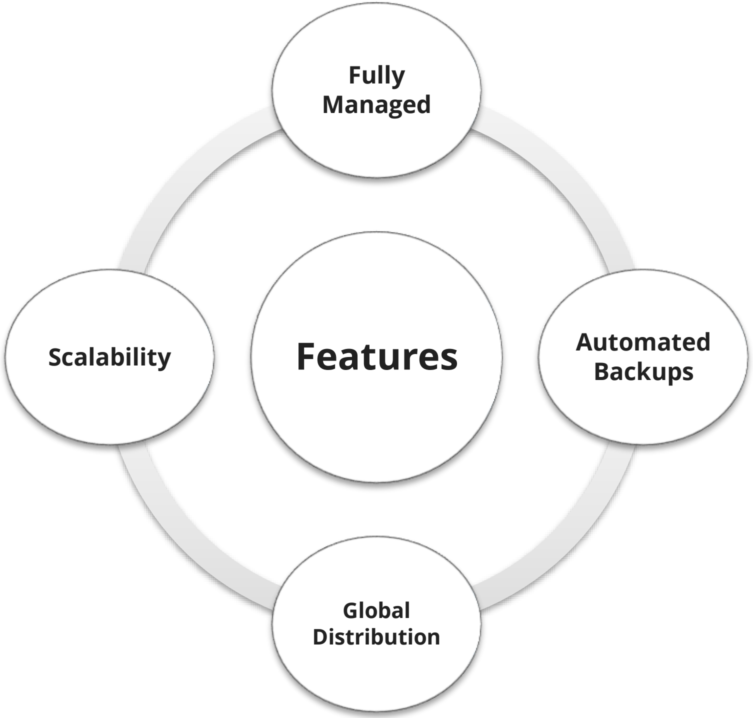
b. Storing data in a denormalized format

[✓ **c. Distributing data across multiple servers for horizontal scalability**]

d. Managing data replication for high availability

Overview of MongoDB Atlas

- MongoDB Atlas is a cloud-based database service provided by MongoDB.
- It is designed to make it easier to deploy, manage, and scale MongoDB databases.



Features

Fully Managed

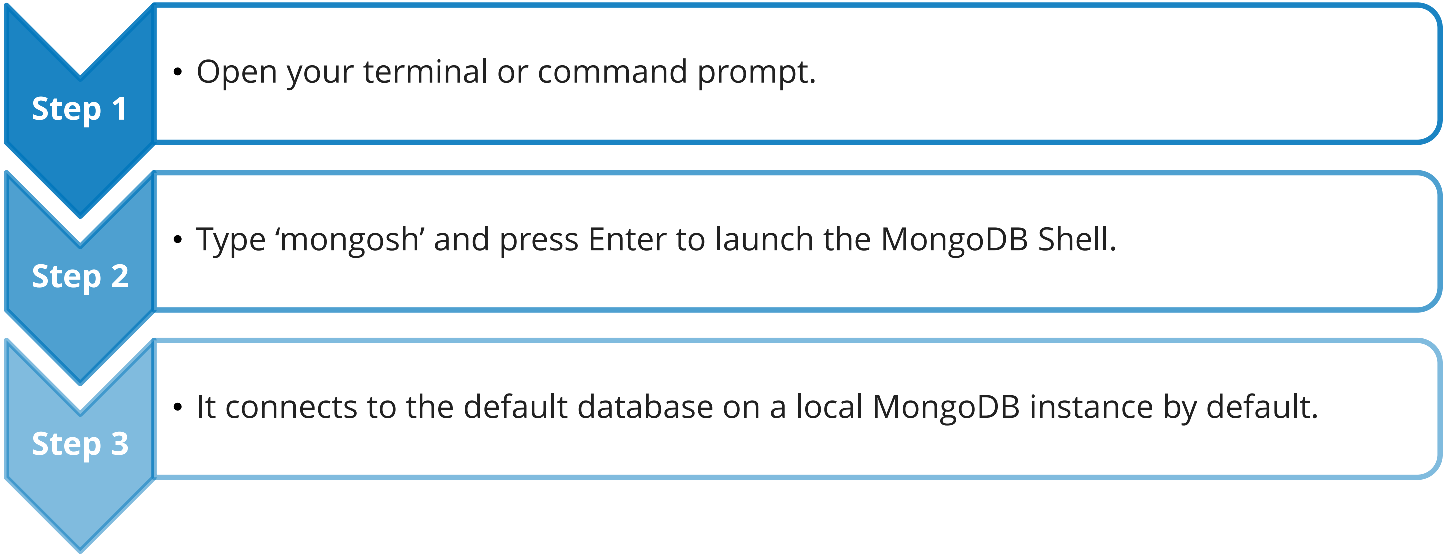
Automated Backups

Global Distribution

Scalability

Using the Shell

- MongoDB Shell is a command-line interface for interacting with MongoDB databases.
- It provides a powerful way to manage data, perform queries, and execute administrative tasks.



Step 1

- Open your terminal or command prompt.

Step 2

- Type ‘mongosh’ and press Enter to launch the MongoDB Shell.

Step 3

- It connects to the default database on a local MongoDB instance by default.

Demo: Using MongoDB Shell

Poll Time

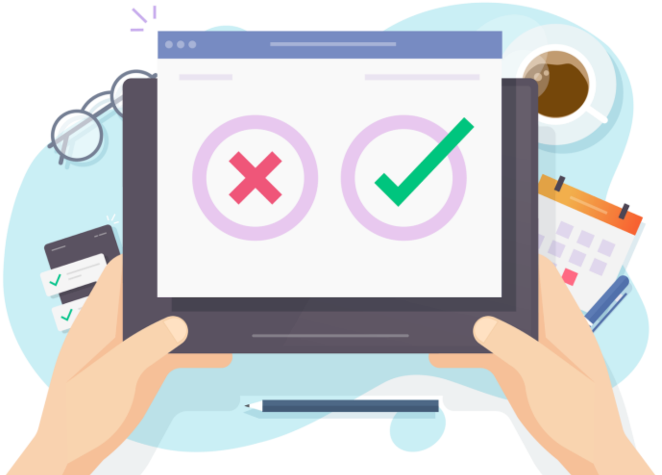
Q. Which MongoDB query operation is used to retrieve all documents from a collection?

a. find({})

b. queryAll()

c. fetchAll()

d. select(*)



Poll Time

Q. Which MongoDB query operation is used to retrieve all documents from a collection?

[✓ a. **find({})**]

b. queryAll()

c. fetchAll()

d. select(*)

Q&A

Activity 1

Pre-requisites:

- Computers with Internet access
- MongoDB Atlas account (or a local MongoDB installation)

Scenario:

You are getting started with MongoDB. Perform the below steps to begin your journey:

- Sign up for a MongoDB Atlas account.
- Create a new MongoDB cluster within MongoDB Atlas.
- Insert the **Car** data into a new collection using the MongoDB Shell.
- Retrieve all documents from the above collection and query based on the below fields.

1. Get a list of all cars with more than 100 bhp of horsepower.
2. Get a list of cars with 4-cylinder engines.
3. Get a list of cars with fuel efficiency of more than 15 mpg.

Summary

- ✓ NoSQL stands for **Not Only SQL**.
- ✓ It represents a diverse set of database technologies designed to overcome the limitations of traditional relational databases.
- ✓ NoSQL databases are categorized into document, key-value, graph, and columnar categories..
- ✓ MongoDB is a leading document-oriented NoSQL database.

Next Session:
CRUD Operations in MongoDB

THANK YOU!

Please complete your assessments and review the self-learning content for this session on the **PRISM** portal.

knowledgehut
upGrad

knowledgehut
upGrad

CRUD Operations in MongoDB

Pre-requisites

Hope you have gone through the self-learning content for this session on the PRISM portal.

By the End of this Session:

- Perform CRUD operation using MongoDB
- Create databases and collections
- Retrieve and filter data from a collection
- Update existing data in a collection
- Delete data from a collection

What have we learned so far?

- NoSQL database allows us to store unstructured and semi-structured data.
- There are 4 types of data models – key-value, document, column, and graph.
- MongoDB belongs is a Document type NoSQL database.
- MongoDB atlas is an online database as a service tool.
- MongoDB compass is GUI that helps in working with the database.

Poll Time

Q. What is the primary advantage of using MongoDB Atlas?

a. Advanced aggregation framework
b. Integration with SQL databases
c. Cloud-based managed service
d. Only supports vertical scaling

Poll Time

Q. What is the primary advantage of using MongoDB Atlas?

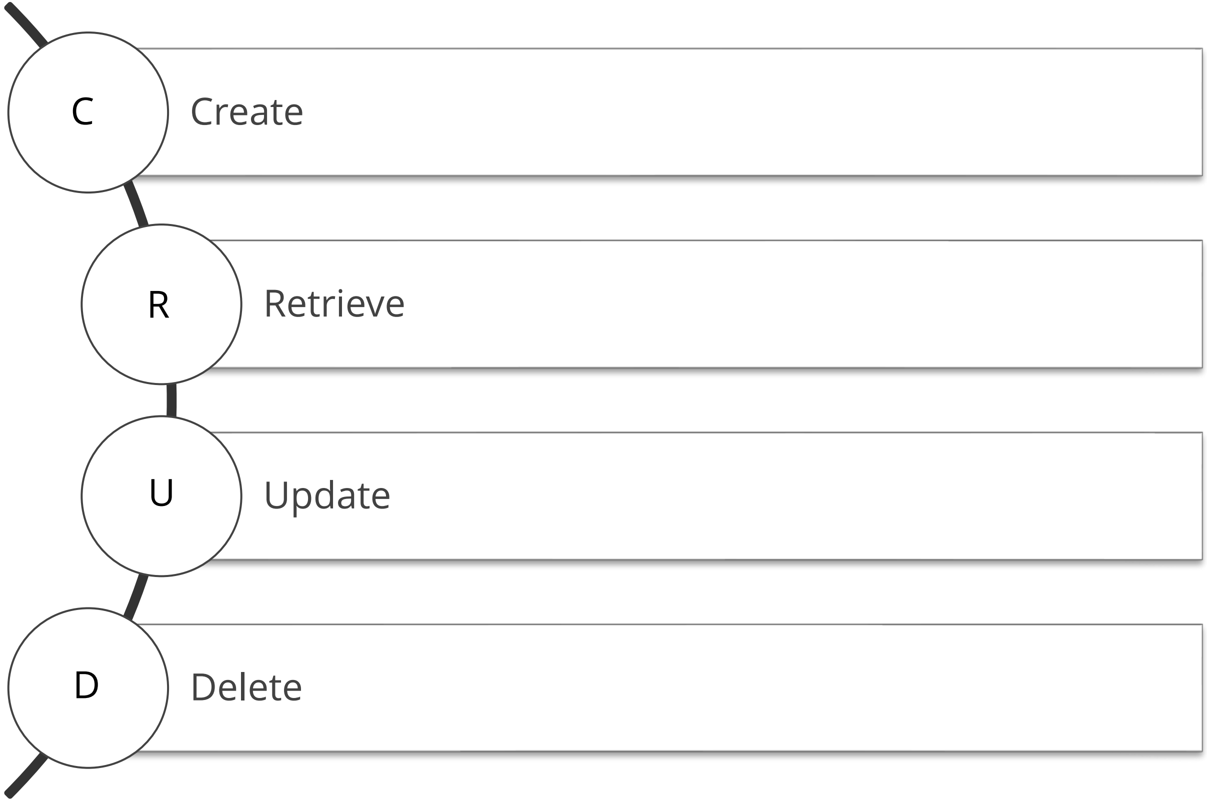
a. Advanced aggregation framework

b. Integration with SQL databases

[✓ c. **Cloud-based managed service**]

d. Only supports vertical scaling

Introduction to CRUD Operations



C Create

R Retrieve

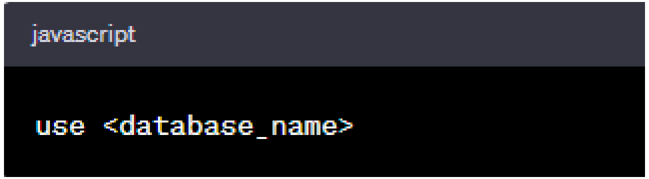
U Update

D Delete

Database and Collection Creation

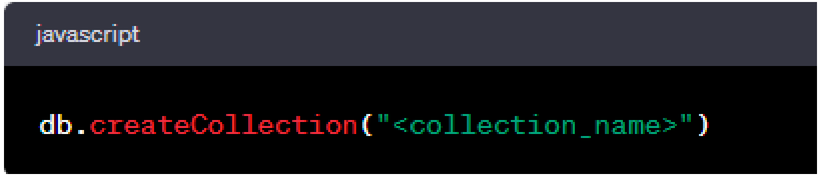
- Databases are containers for collections in MongoDB.
- Each database can hold multiple collections with related data.
- Collections are containers for storing MongoDB documents.
- Each database can contain multiple collections to organize related data.

Creating a Database: Syntax



```javascript
use <database_name>
```

Creating a Collection: Syntax

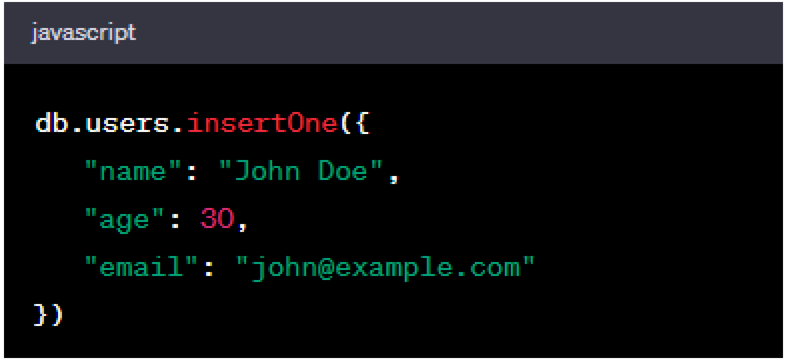


```javascript
db.createCollection("<collection_name>")
```

Document Insertion

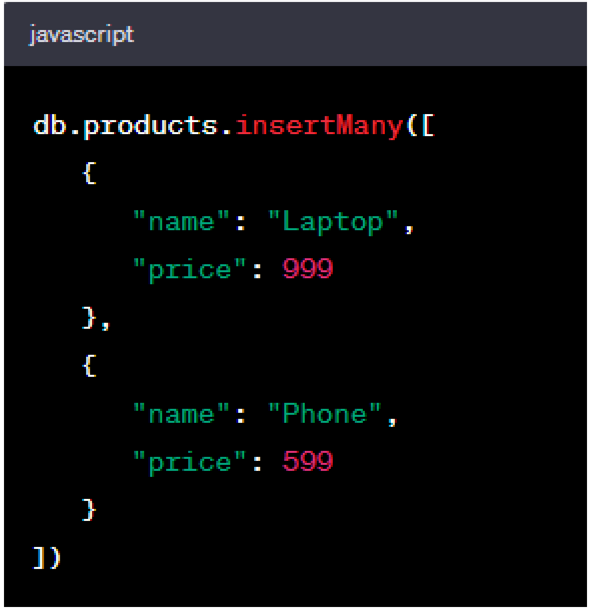
- Document insertion is a fundamental operation in MongoDB.
- It involves adding data records (documents) to a collection.
- Use the **insertOne()** method to insert a single document.
- Use the **insertMany()** method to insert multiple documents.

Example: Inserting a Document



```javascript
db.users.insertOne({
  "name": "John Doe",
  "age": 30,
  "email": "john@example.com"
})
```

Example: Inserting Multiple Documents

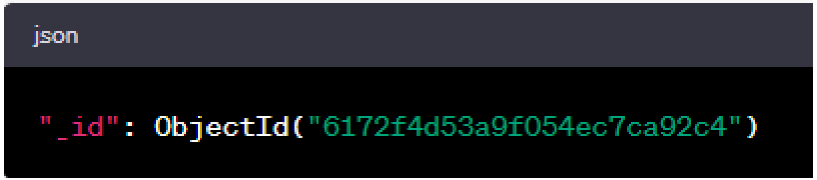


```javascript
db.products.insertMany([
  {
    "name": "Laptop",
    "price": 999
  },
  {
    "name": "Phone",
    "price": 599
  }
])
```

The Unique _id Field

- The _id field is a crucial component in MongoDB documents.
- It uniquely identifies each document within a collection.
- If not provided, MongoDB automatically generates a unique _id for each document.
- You can provide your own unique _id values during document insertion.

Example: ObjectId



```json
"_id": ObjectId("6172f4d53a9f054ec7ca92c4")
```

Pop Quiz

Q. What is the primary purpose of the _id field in MongoDB documents?

a. It defines the data type of the document
b. It specifies the collection in which the document is stored
c. It uniquely identifies each document within a collection
d. It determines the indexing strategy for the document

Pop Quiz

Q. What is the primary purpose of the _id field in MongoDB documents?

a. It defines the data type of the document

b. It specifies the collection in which the document is stored

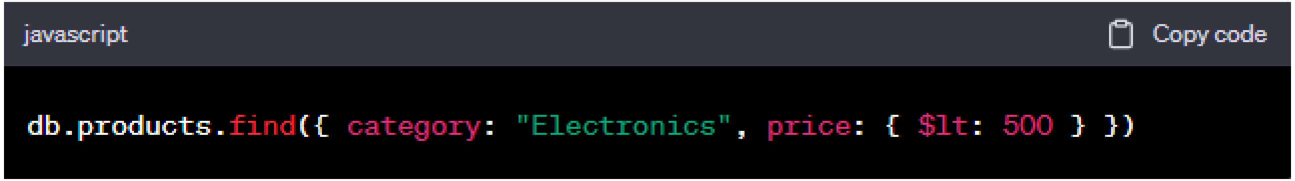
[✓ **c. It uniquely identifies each document within a collection**]

d. It determines the indexing strategy for the document

Filtering and Querying Documents

- Filtering and querying are essential for retrieving specific data from MongoDB collections.
- The **find()** method is used to retrieve documents from a collection.
- MongoDB offers various query operators for advanced filtering.
- Examples: **$eq, $gt, $lt, $in, $and, $or**.

Example Query: Find Documents

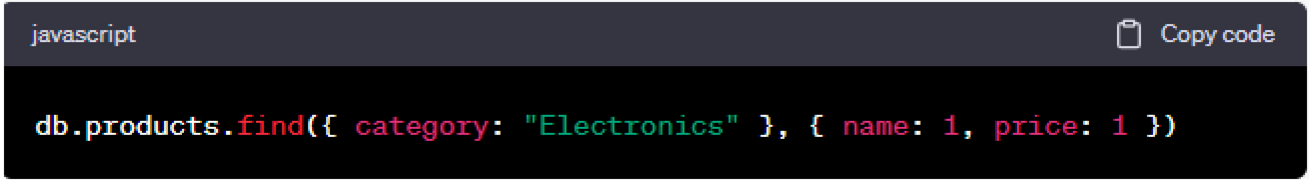


```javascript
db.products.find({ category: "Electronics", price: { $lt: 500 } })
```

Projection in the find Method

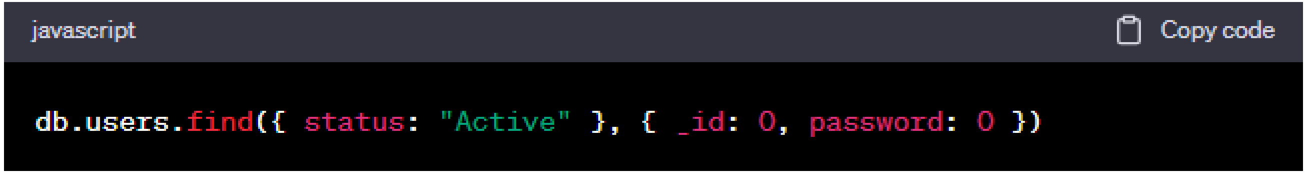
- Projection allows you to control which fields are included or excluded in query results.
- Fine-tune the data returned from MongoDB queries.
- Use the second argument of the find() method for projection.
- Specify fields to include (1) or exclude (0).

Example Projection: Including Fields



```javascript
db.products.find({ category: "Electronics" }, { name: 1, price: 1 })
```

Example Projection: Excluding Fields



```javascript
db.users.find({ status: "Active" }, { _id: 0, password: 0 })
```

Demo

Poll Time

Q. Which MongoDB query syntax is used for filtering documents with a specific field value?

a. db.collection(filter)

b. db.collection.find({ field: value })

c. db.collection.query({ field: value })

d. db.collection.retrieve({ field: value })

Poll Time

Q. Which MongoDB query syntax is used for filtering documents with a specific field value?

a. db.collection(filter)

▌**b. db.collection.find({ field: value })**

c. db.collection.query({ field: value })

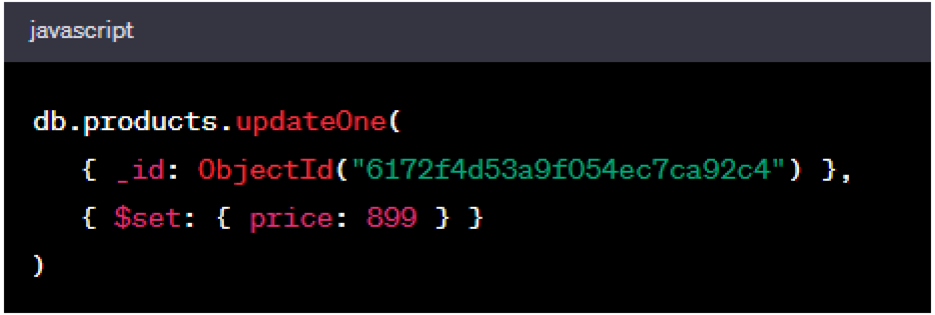
d. db.collection.retrieve({ field: value })

Q&A
?
?
?
?
?
?
?
?

Updating Documents

- Use the **updateOne()** method to update a single document.
- Use the **updateMany()** method to update multiple documents.
- The **$set** operator sets the value of a field.

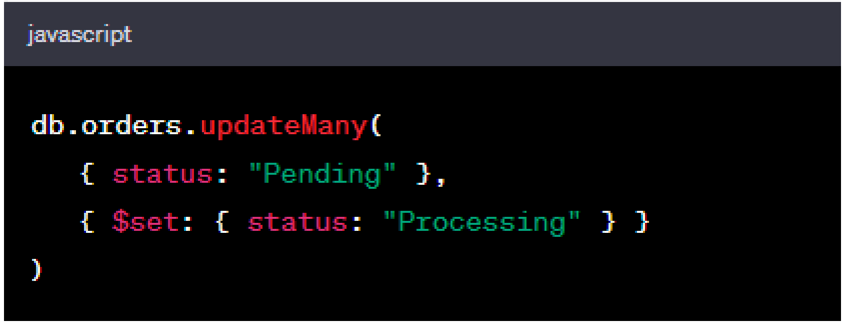
Example Update: updateOne()



javascript

```javascript
db.products.updateOne(
   { _id: ObjectId("6172f4d53a9f054ec7ca92c4") },
   { $set: { price: 899 } }
)
```

Example Update: updateMany()



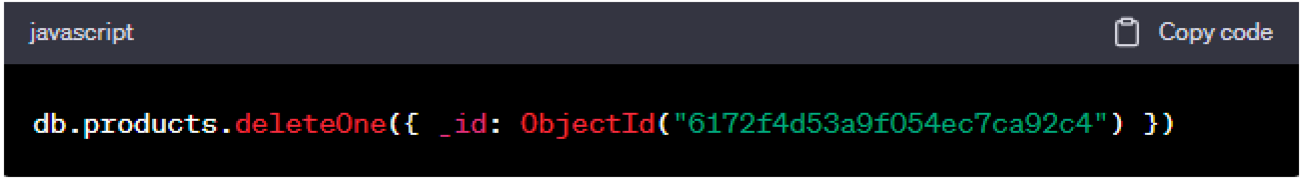
javascript

```javascript
db.orders.updateMany(
   { status: "Pending" },
   { $set: { status: "Processing" } }
)
```

Deleting Documents

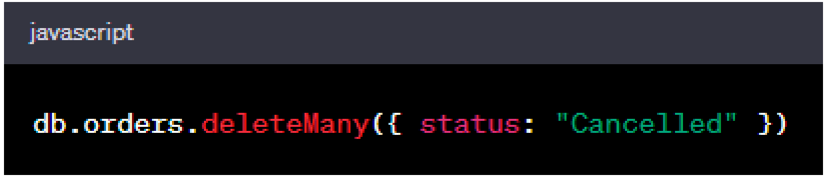
- Use the **deleteOne()** method to remove a single document.
- Use the **deleteMany()** method to remove multiple documents.
- Specify the **_id** field in the filter to delete by unique identifier.

Example Delete: deleteOne()



```javascript
db.products.deleteOne({ _id: ObjectId("6172f4d53a9f054ec7ca92c4") })
```

Example Delete: deleteMany()

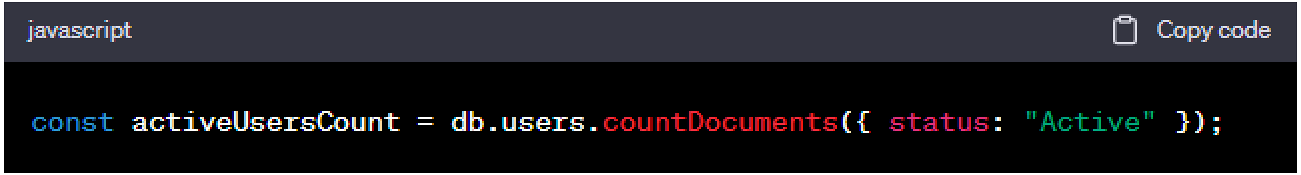


```javascript
db.orders.deleteMany({ status: "Cancelled" })
```

Counting Documents

- Counting documents is a common operation for data analysis and validation.
- Use the **countDocuments()** method to count matching documents.

Example Count: Matching Documents

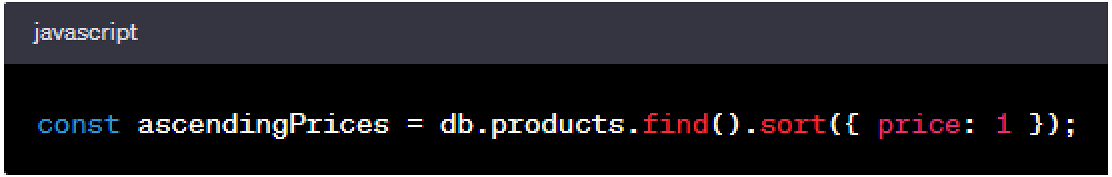


```javascript
const activeUsersCount = db.users.countDocuments({ status: "Active" });
```

Sorting and Limiting Results

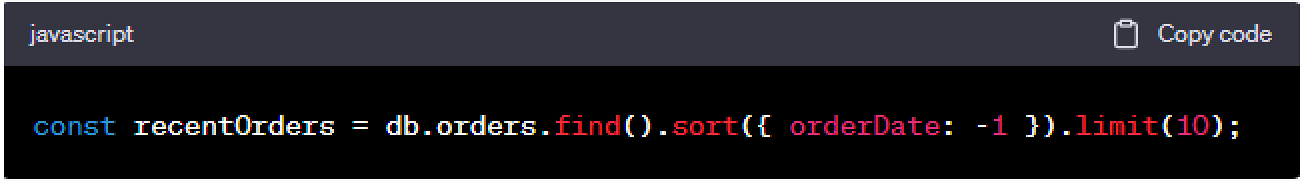
- Use the **sort()** method to order query results.
- Specify 1 against the field name to sort in ascending order and -1 to sort in descending order.
- . Use the **limit()** method to restrict the number of returned documents.

Example Sort: Ascending



```javascript
const ascendingPrices = db.products.find().sort({ price: 1 });
```

Example Limit



```javascript
const recentOrders = db.orders.find().sort({ orderDate: -1 }).limit(10);
```

Pop Quiz

Q. In MongoDB, which value is used to specify ascending sorting order when using the sort() method?

a. 0

b. "asc"

c. 1

d. "ascending"

Pop Quiz

Q. In MongoDB, which value is used to specify ascending sorting order when using the sort() method?

a. 0

b. "asc"

[✓ **c. 1**]

d. "ascending"

Demo

Q&A
?
?
?
?
?
?
?
?

Activity 1

Pre-requisites:

- Computers with internet access
- MongoDB Atlas account (or a local MongoDB installation)
- MongoDB Shell or MongoDB Compass (optional, for GUI interaction)

Scenario:

You are getting started with MongoDB. Perform the below steps to begin your journey:

- Use the car collection which was created in the last activity and perform the below operation on the same:
  - Insert a new car with the below specifications.

```
{
        “Name”: Volkswagen Polo,
        “Cylinders”: 4,
        “Year”: 2013
}
```

Activity 2

Scenario:

Continue the previous activity and perform the below operations:

- Update cars with 4 cylinders to have 125 bhp horsepower.
- Delete all cars with more than 200 cc of displacement.
- Retrieve all cars which are manufactured after 1975 and sort the results in the ascending order of horsepower.
- Limit the number of results of the previous query to 10.

Summary

- ✓ CRUD operation allow us to manipulate existing data in a collection.
- ✓ Filtering and querying retrieve specific data from MongoDB collections.
- ✓ Use updateOne() for single updates and updateMany() for multiple updates.
- ✓ Use deleteOne() for single deletions and deleteMany() for multiple deletions.
- ✓ Use sort() to order documents by field values.
- ✓ Employ limit() to restrict the number of returned documents.

Session Feedback

Next Session:
MongoDB Drivers and Python

THANK YOU!

Please complete your assessments and review the self-learning content for this session on the **PRISM** portal.

knowledgehut
upGrad