

**Prof R. Madana Mohana**



# **BIG DATA ANALYTICS**

## **MongoDB**

### **Overview**

<https://www.youtube.com/c/RASINENIMADANAMOHANA>



# MongoDB: *Reference Material*

---

**Tutorialspoint:**

<https://www.tutorialspoint.com/mongodb/index.htm>

# MongoDB - Overview

# MongoDB: *Overview*

- **MongoDB** is an **open-source** document database and leading **NoSQL** database.
- **MongoDB** is written in **C++**.
- **MongoDB** is a **cross-platform**, document oriented database that provides, **high performance**, **high availability**, and **easy scalability**.
- **MongoDB** works on concept of **collection** and **document**.

# MongoDB: *Overview*

## Database:

- **Database** is a physical container for collections.
- Each database gets its own set of files on the file system.
- A single **MongoDB** server typically has multiple databases.

# MongoDB: *Overview*

## Collection:

- **Collection** is a **group** of **MongoDB** documents.
- It is the equivalent of an **RDBMS** table.
- A **collection** exists within a **single** database.
- **Collections** do not enforce a **schema**.
- **Documents** within a **collection** can have **different** fields.
- Typically, **all** documents in a **collection** are of **similar** or **related** purpose.

# MongoDB: *Overview*

## Document:

- A **document** is a set of **key-value** pairs.
- **Documents** have **dynamic schema**.
- **Dynamic schema** means that **documents** in the **same collection** do not need to have the **same set of fields** or **structure**, and **common fields** in a collection's **documents** may hold **different types of data**.

# MongoDB: *Overview*

---

## Document:

- The *following table* shows the relationship of **RDBMS** terminology with **MongoDB**.



# MongoDB: *Overview*

## Document:

RDBMS	MongoDB
Database	Database
Table	Collection
Tuple/Row	Document
column	Field
Table Join	Embedded Documents
Primary Key	Primary Key (Default key _id provided by mongodb itself)
<b>Database Server and Client</b>	
Mysqld/Oracle	mongod
mysql/sqlplus	mongo

# MongoDB: *Overview*

## Sample Document:

- Following example shows the document structure of a blog site, which is simply a *comma separated key-value* pair.

# MongoDB: *Overview*

## Sample Document:

```
{
  _id: ObjectId(7df78ad8902c)
  title: 'Working mechanisms of Mongo DB',
  description: 'MongoDB is no sql database',
  by: 'BIG DATA ANALYTICS',
  url: 'https://www.mongodb.com/docs/manual/',
  tags: ['mongodb', 'database', 'NoSQL'],
  likes: 100,
```



# MongoDB: *Overview*

## Sample Document:

```
comments: [  
  {  
    user: 'user1',  
    message: 'My first comment',  
    dateCreated: new Date(2022, 9, 28, 2, 15),  
    like: 0  
  },
```

# MongoDB: *Overview*

## Sample Document:

```
{
  user: 'user2',
  message: 'My second comments',
  dateCreated: new Date(2022, 9, 29, 7, 45),
  like: 5
}
```

# MongoDB: *Overview*

## Sample Document:

- **\_id** is a **12 bytes hexadecimal number** which assures the **uniqueness of every document**.
- We can provide **\_id** while **inserting** the **document**.
- If we don't provide then **MongoDB** provides a **unique id** for **every document**.
- These **12 bytes** first 4 bytes for the **current timestamp**, next 3 bytes for **machine id**, next 2 bytes for **process id** of **MongoDB server** and remaining 3 bytes are simple **incremental VALUE**.