

A large yellow hexagon with a black grid pattern of dots and lines, serving as a background for the title text.

# INTRODUCTION

## Mongo DB

A smaller blue hexagon with a white background, containing the text "Data Sheet".

Data  
Sheet

# Introduction to MongoDB

MongoDB is a document-oriented NoSQL database used for data storage of high volumes. MongoDB uses lists and records instead of using tables and rows as in conventional relational databases. Documents consist of key-value pairs that are MongoDB's basic data device. Collections include collections of documents and functions that are the equivalent of tables of relational databases.

## MongoDB Features

- ❖ The collections found in each database are documents in turn. With a varying number of fields, every document may be different. Each document's size and content can vary from each other.
- ❖ The structure of the document is more in line with how developers construct their classes and objects in their respective programming languages. Developers would also say that their groups are not rows and columns, but with key-value pairs providing a simple structure.
- ❖ The rows (or documents, as called in MongoDB) need not have a schema specified in advance.
- ❖ With MongoDB's data model, you can more easily represent hierarchical relations, store arrays and other complex structures.

## Data Modelling in MongoDB

Keep the following things in mind while modelling data in Mongo:

- ❖ What are the applications requirements – look at the application's business needs to see what data to form of data the application requires. On this basis, ensure that the document structure is accordingly defined.
- ❖ What are data recovery trends – If you expect the use of heavy queries, consider using indexes in your data model to maximize query performance.
- ❖ Are the servers regularly used for inserts, changes and deletions? To enhance the efficiency of the entire MongoDB setting, consider using indexes or implement sharding as needed in your data modelling design.



## MongoDB Architecture's Main Components

**id** - This is a field that any MongoDB document needs. A unique value in a MongoDB document is represented by the id field.

**Collection** - This is a grouping of the documents from MongoDB. A set is the equivalent of a table that every other RDMS, such as Oracle or MS SQL, produces.

**Cursor** - This is a reference to a query's result collection. Clients will iterate to get results by means of a cursor.

**Database** - This is a collection container as in RDMS, in which it is a tables container. Each database on the filesystem gets its own collection of files. Multiple databases can be handled by a MongoDB server.

**Document** - Basically, a record is called a document in a MongoDB set. The document would, in turn, consist of the name of the field and the values.

**Field** - In relational databases, fields are similar to columns.

**Json** - This is known as Object Notation JavaScript. This is a plain text format, human-readable for communicating structured data. JSON is provided in several programming languages at the moment.