Module 3
Chapter 6 – Introduction to MongoDB

Prepared By:
Mr. Rajesh Nayak
Department of Artificial Intelligence and Data Science

- **6.1 What is MongoDB?**
- MongoDB is (i) Cross-platform, (ii) Open Source, (iii) Non-relational, (iv) Distributed, (v) NoSQL, (vi) Document-oriented data store.
- **6.2 Why MongoDB?**
- Few of the major challenges with traditional RDBMS are dealing with large volumes of data, rich variety of data, and meeting up to the scale needs of enterprise data.
- The need is for a database that can scale out or scale horizontally to meet the scale requirements, has flexibility with respect to schema, is fault tolerant, is consistent, and partition tolerant, and can be easily distributed over a multitude of nodes in a cluster.

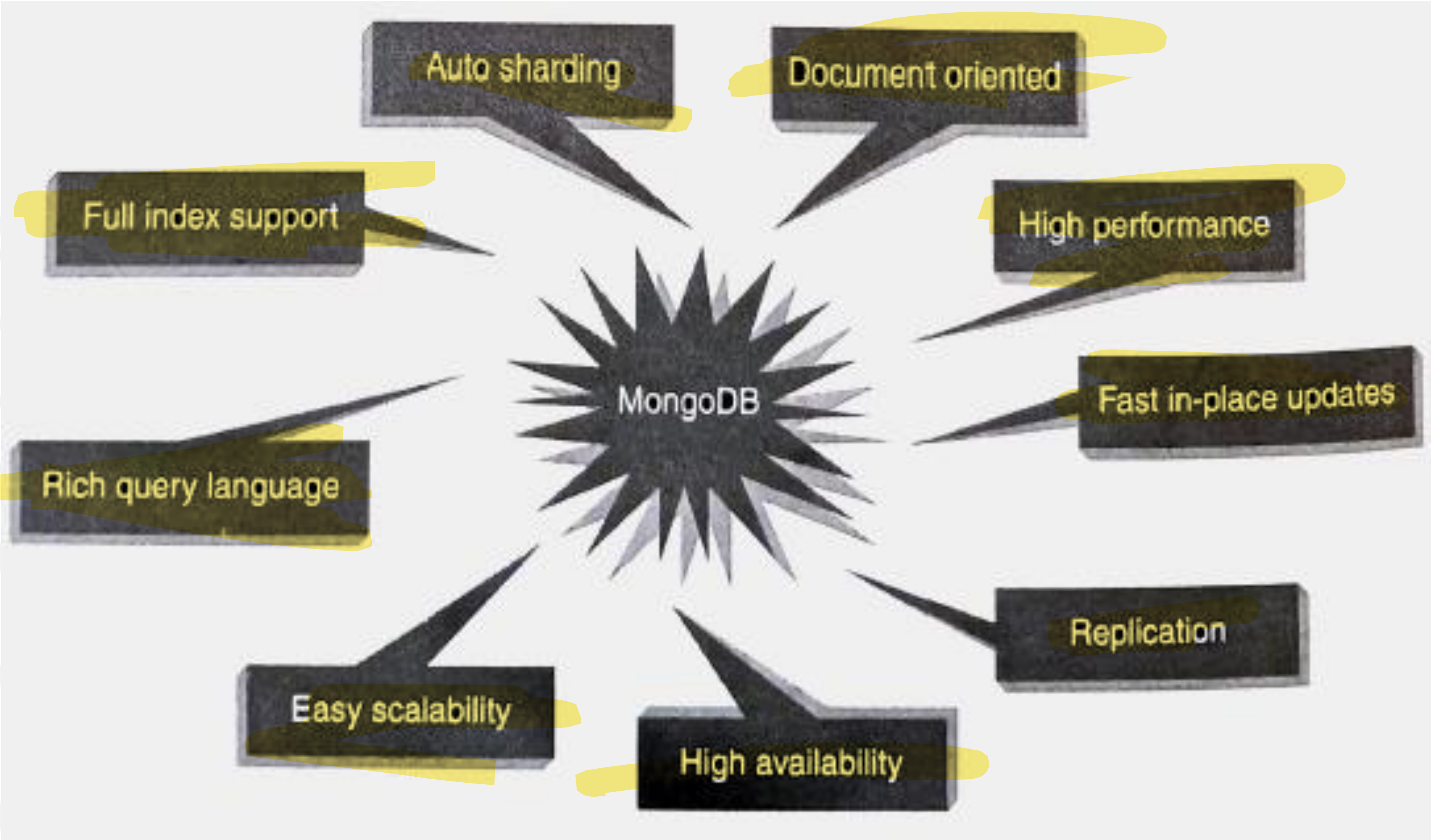
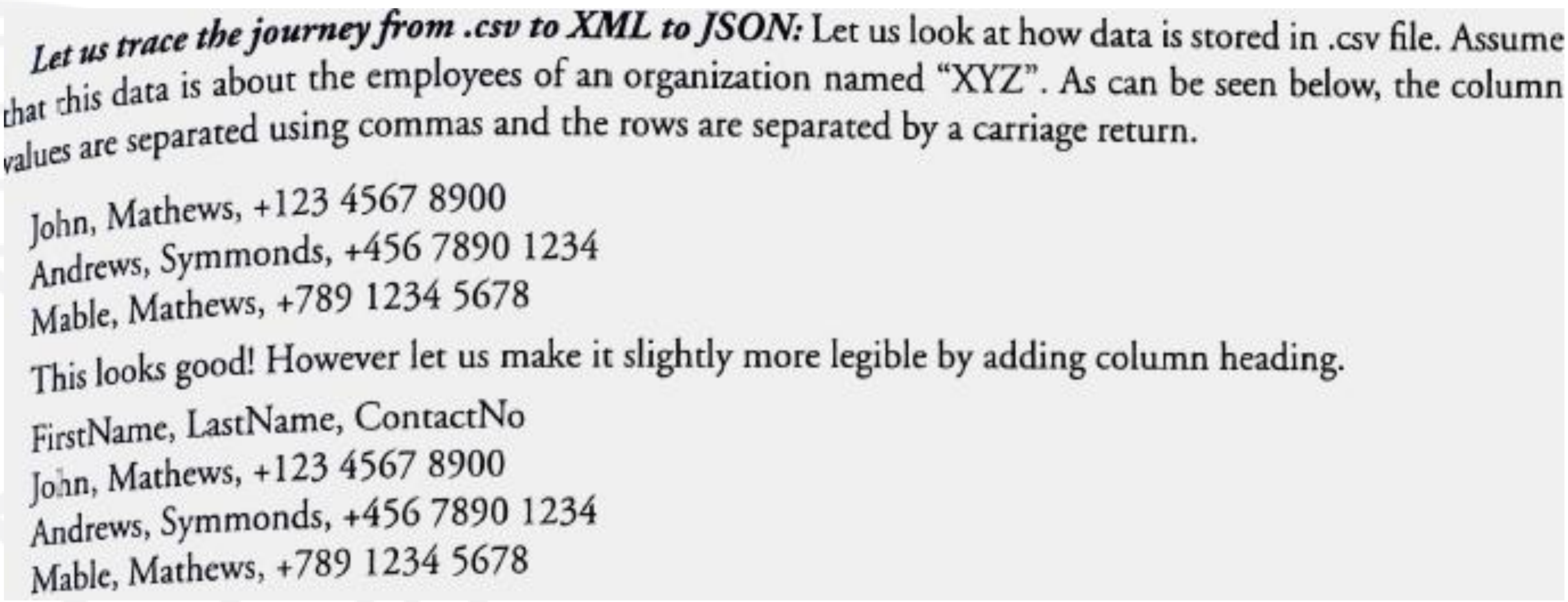


Figure 6.1 Why MongoDB?

- **6.2.1 Using Java Script Object Notation (JSON)**
- MongoDB actually does not use JSON but BSON – it is Binary JSON. It is an open standard. It is used to store complex data structures.



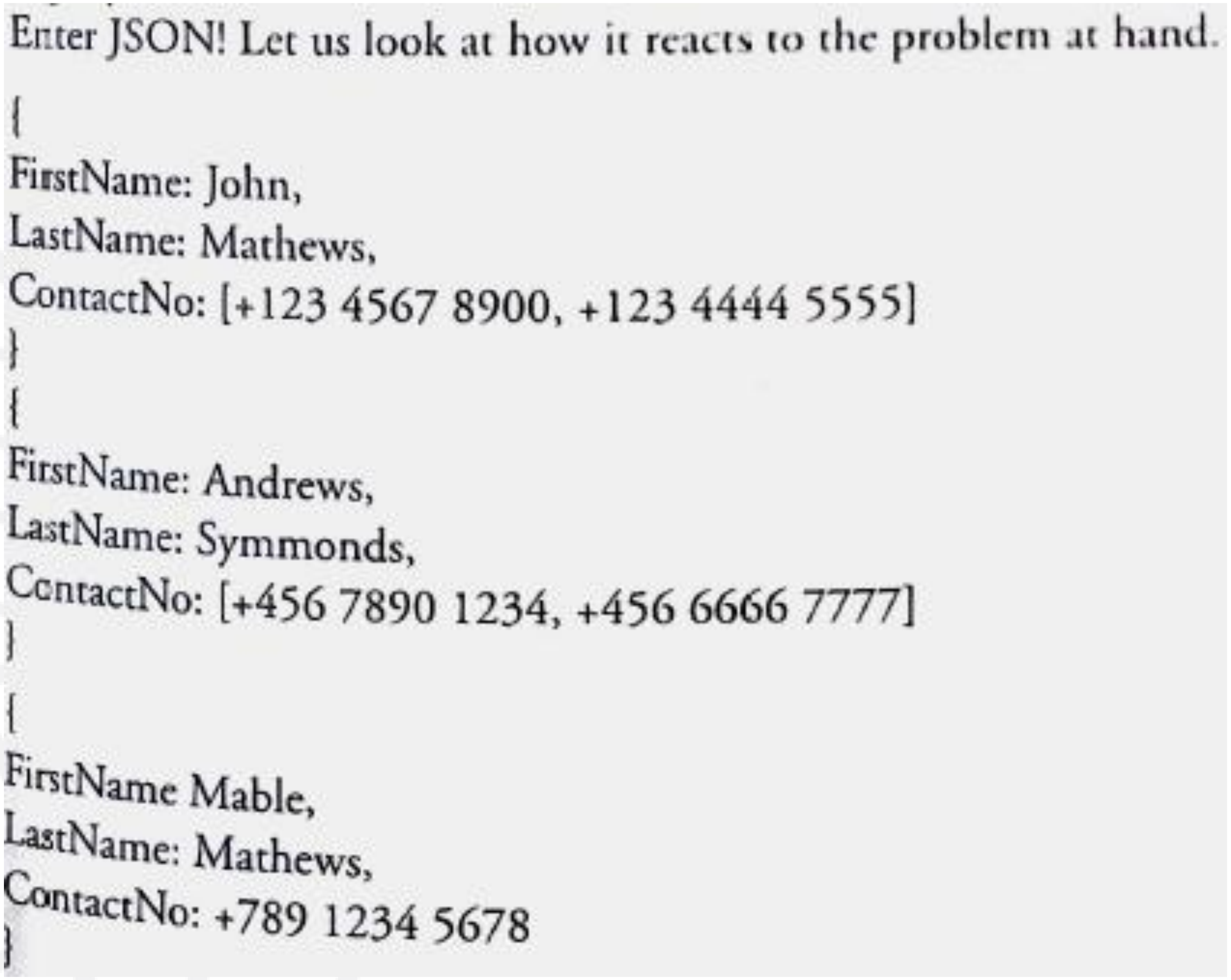
Let us trace the journey from .csv to XML to JSON: Let us look at how data is stored in .csv file. Assume that this data is about the employees of an organization named "XYZ". As can be seen below, the column values are separated using commas and the rows are separated by a carriage return.

```
John, Mathews, +123 4567 8900
Andrews, Symmonds, +456 7890 1234
Mable, Mathews, +789 1234 5678
```

This looks good! However let us make it slightly more legible by adding column heading.

```
FirstName, LastName, ContactNo
John, Mathews, +123 4567 8900
Andrews, Symmonds, +456 7890 1234
Mable, Mathews, +789 1234 5678
```

- **6.2.1 Using Java Script Object Notation (JSON)**

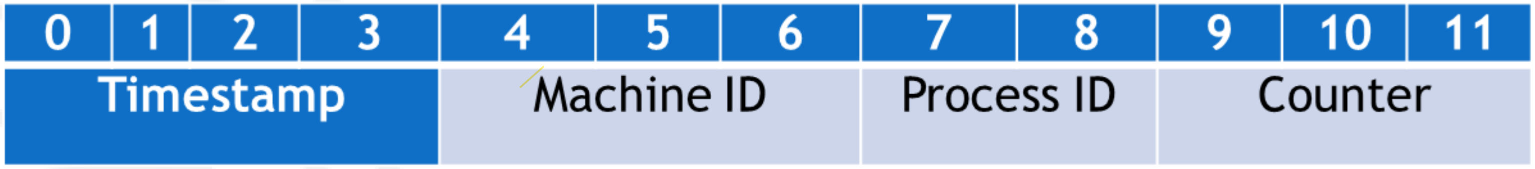


Enter JSON! Let us look at how it reacts to the problem at hand.

```
{
FirstName: John,
LastName: Mathews,
ContactNo: [+123 4567 8900, +123 4444 5555]
}
{
FirstName: Andrews,
LastName: Symmonds,
ContactNo: [+456 7890 1234, +456 6666 7777]
}

{
FirstName Mable,
LastName: Mathews,
ContactNo: +789 1234 5678
}
```

- **6.2.2 Creating or Generating a Unique Key**
- Each JSON document should have a unique identifier. It is the _id key similar to primary key in relational databases.
- This facilitates search for documents based on the unique identifier.
- An index is automatically built on the unique identifier. Either user can provide the unique value or the Mongo shell will generate the same.



- **Database:** It’s a collection of collections. It gets created the first time user collection makes reference to it. This can also be created on demand. Each database gets its own set of files on the file system. Single MongoDB server can have several databases.

- **Collection:** A collection is analogous to a table of RDBMS. A collection is created on demand. Its created the first time when user attempt to save a document that references it. A collection exists within a single database. It holds several MongoDB documents.
- Collection does not enforce a schema. This implies that documents within a collection can have different fields.
- **Document:** A document is analogous to a row/record/tuple in an RDBMS table. A document has a dynamic schema.
- This implies that a document in a collection need not necessarily have the same set of fields/key-value pairs.

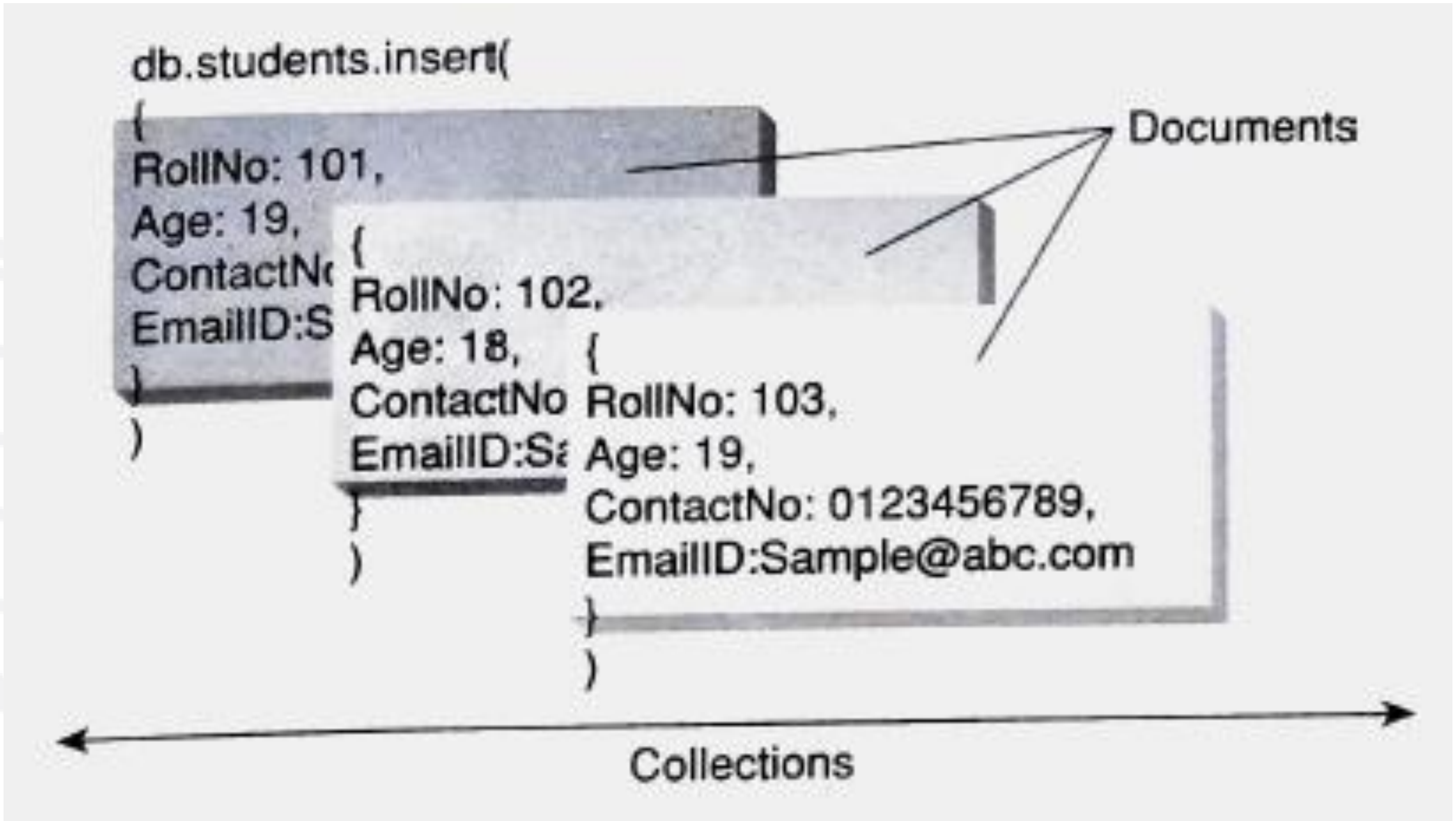
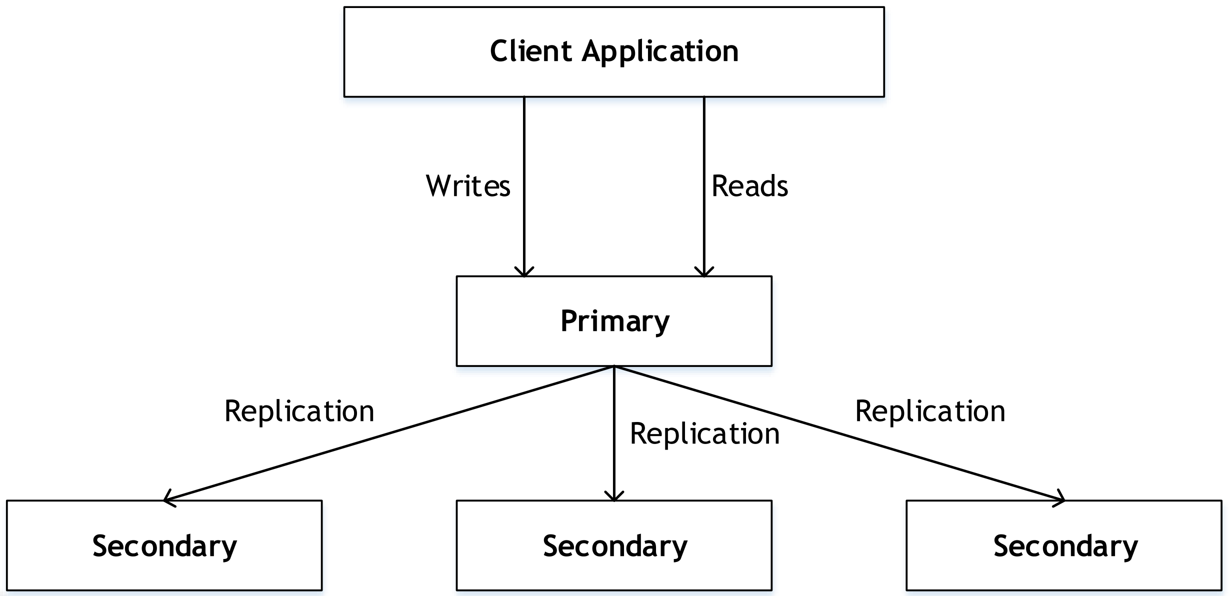


Figure 6.2 A collection "students" containing 3 documents.

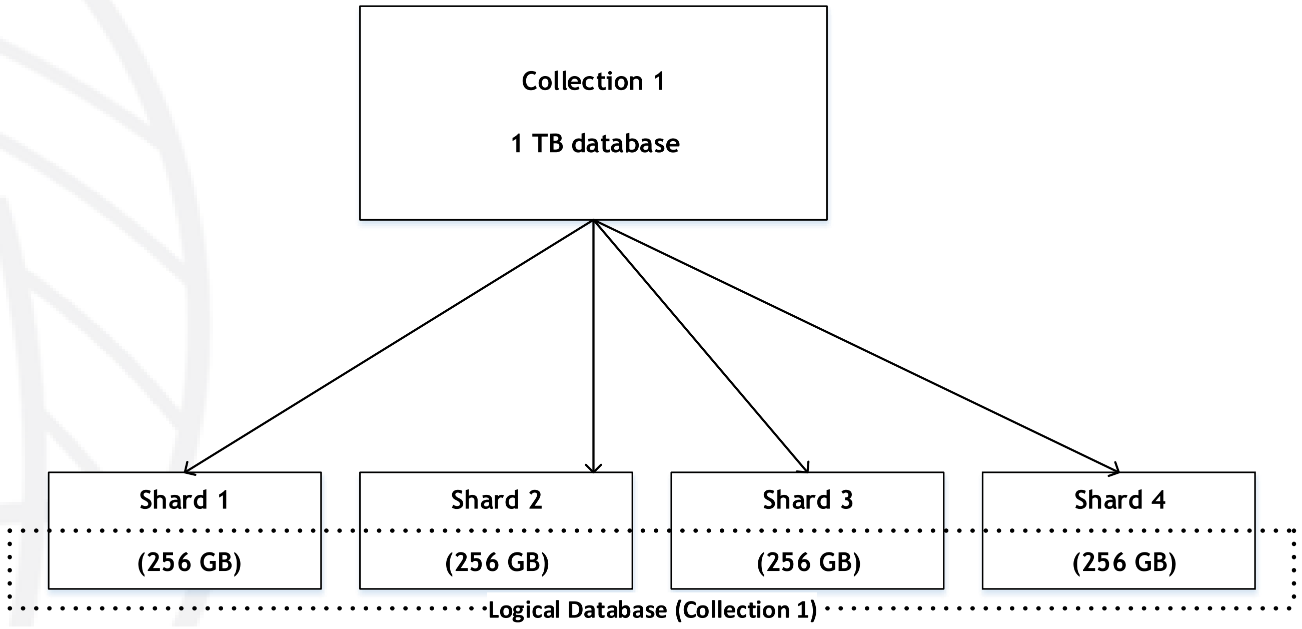
- **6.2.3 Support for Dynamic Queries**
- MongoDB has extensive support for dynamic queries.
- This is in keeping with traditional RDBMS wherein we have static data and dynamic queries.

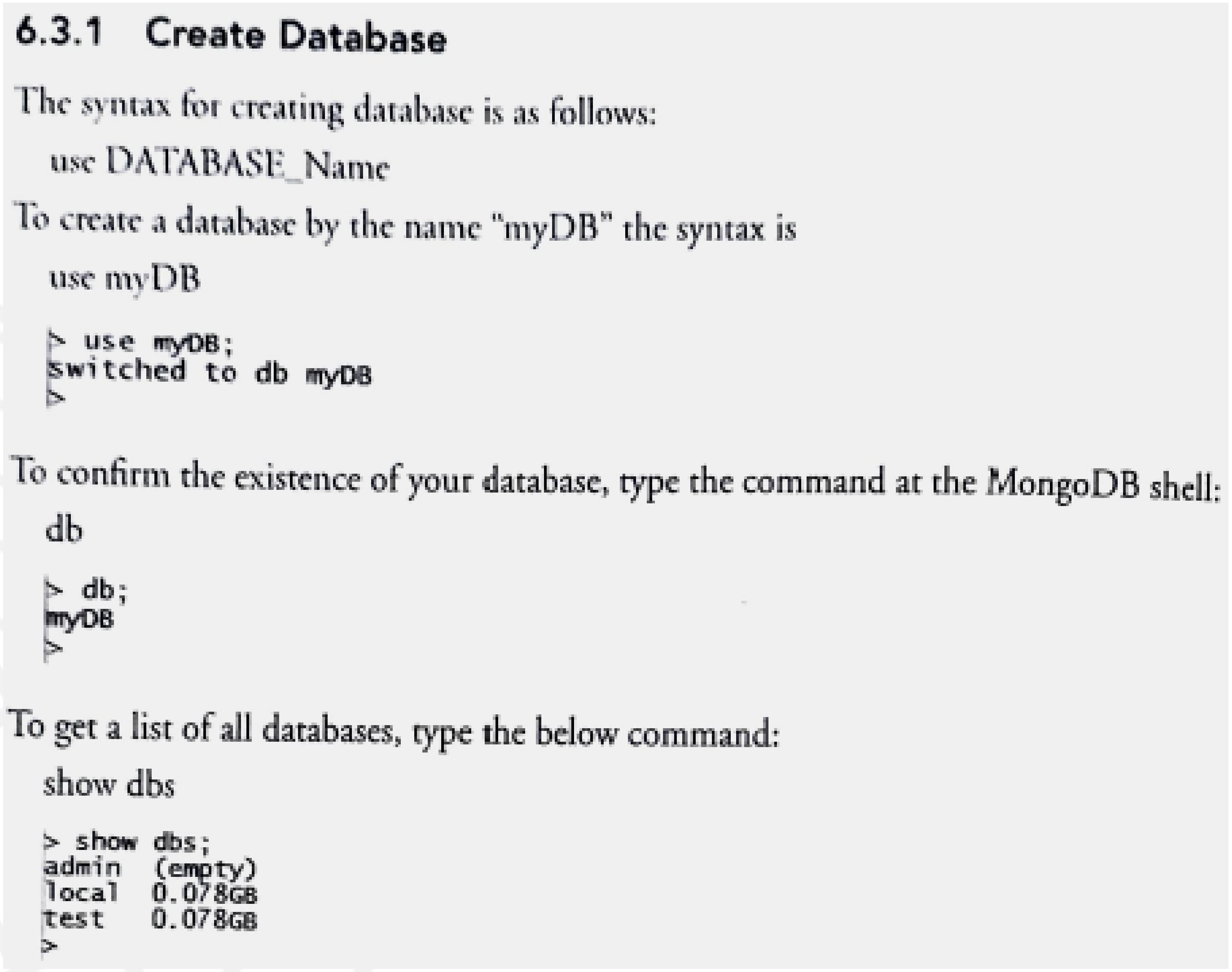
- **6.2.4 Storing Binary Data**
- MongoDB provides GridFS to support the storage of binary data. It can store up to 4 MB. It stores the metadata in a collection called “file”. Then it breaks them into small pieces called “chunks”.
- **6.2.5 Replication**
- **6.2.6 Sharding**
- **6.2.7 Updating Information In-Place**

Replication



Sharding





6.3.1 Create Database

The syntax for creating database is as follows:

use DATABASE_Name

To create a database by the name "myDB" the syntax is

use myDB

```
> use myDB;
switched to db myDB
>
```

To confirm the existence of your database, type the command at the MongoDB shell:

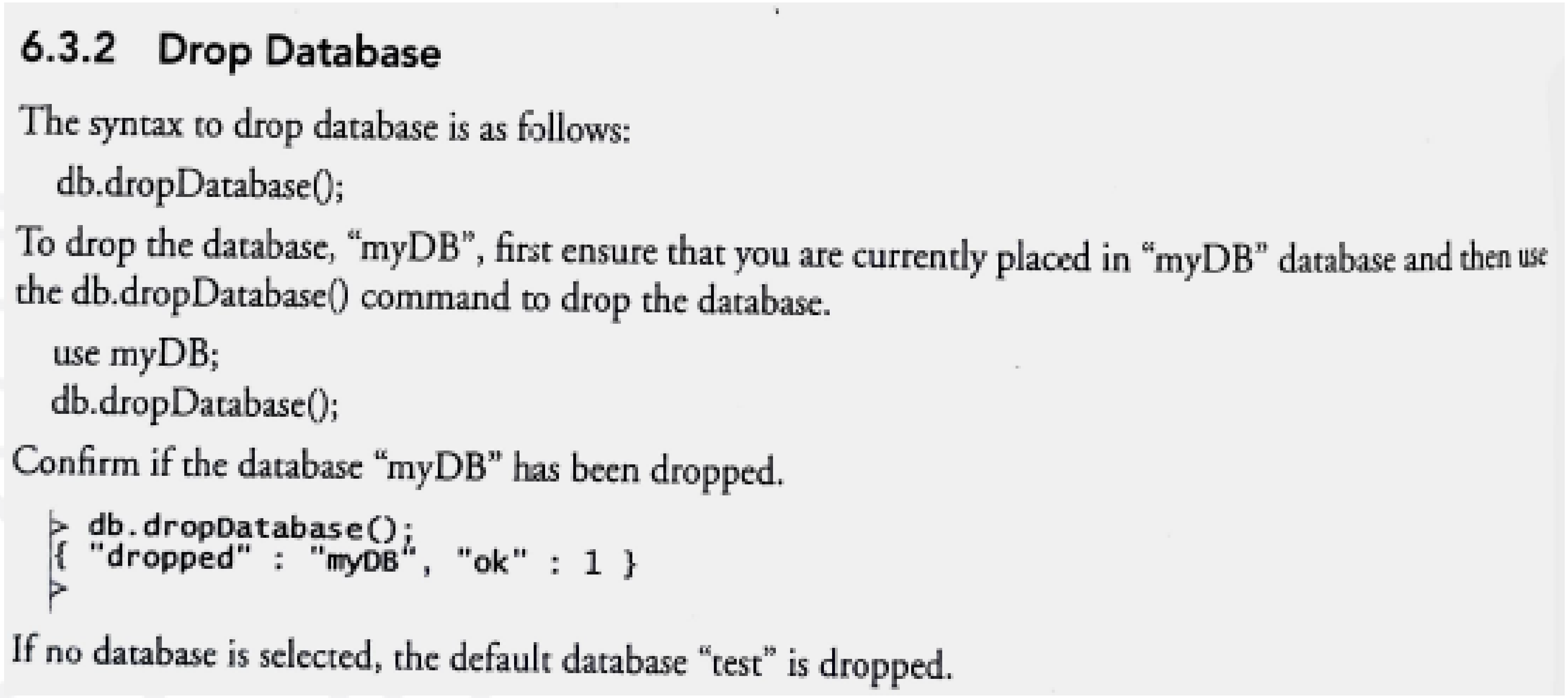
db

```
> db;
myDB
>
```

To get a list of all databases, type the below command:

show dbs

```
> show dbs;
admin   (empty)
local   0.078GB
test    0.078GB
>
```



6.3.2 Drop Database

The syntax to drop database is as follows:

```
db.dropDatabase();
```

To drop the database, "myDB", first ensure that you are currently placed in "myDB" database and then use the db.dropDatabase() command to drop the database.

```
use myDB;
db.dropDatabase();
```

Confirm if the database "myDB" has been dropped.

```
> db.dropDatabase();
{ "dropped" : "myDB", "ok" : 1 }
>
```

If no database is selected, the default database "test" is dropped.