Full-Stack Web Development

Full-Stack viva/interview questions.

1. What is MongoDB?

MongoDB is a NoSQL database that stores data in flexible, JSON-like documents. Unlike traditional relational databases (like MySQL), it does not use tables with a fixed schema. This makes it highly flexible and scalable for handling unstructured or semi-structured data.

2. How is MongoDB different from a traditional SQL database?

The biggest difference lies in their data model.

- **SQL Databases** (e.g., MySQL, PostgreSQL) are **relational**. They store data in **tables** with a rigid, predefined schema. Relationships between tables are established using foreign keys.
- **MongoDB** is **non-relational**. It stores data in **collections** of **documents** and uses a flexible schema. This means documents in the same collection can have different fields. This approach avoids the need for complex table joins, which can make read operations faster.

---

3. What is a "document" in MongoDB?

A document is the basic unit of data in MongoDB. It's a record composed of field and value pairs, similar to a JSON object. Documents can contain nested documents, arrays, and other data types, allowing for rich and complex data structures within a single record.

- **Example:**

JSON

```
{
  "name": "John Doe",
  "age": 30,
  "address": {
    "street": "123 Main St",
    "city": "Anytown"
  },
  "hobbies": ["reading", "hiking"]
}
```

4. What is a "collection" in MongoDB?

A collection is a grouping of MongoDB documents. It's the equivalent of a table in a relational database. A collection resides within a database and can contain multiple documents. The key feature is that, unlike a table, a collection does not enforce a rigid schema, so its documents can have different structures.

---

5. How do you perform CRUD operations in MongoDB?

CRUD stands for Create, Read, Update, and Delete. In MongoDB, these operations are performed using simple shell commands or through driver methods in your application.

- **Create:** Use db.collection.insertOne() or db.collection.insertMany() to add new documents.
- **Read:** Use db.collection.find() to query documents based on specific criteria.
- **Update:** Use db.collection.updateOne() or db.collection.updateMany() to modify existing documents.
- **Delete:** Use db.collection.deleteOne() or db.collection.deleteMany() to remove documents.

This video provides a great visual explanation of what MongoDB collections and documents are.
What is MongoDB Collection and Document