

Slide 1: Introduction

- MongoDB vs. SQL Databases :
- Brief overview of NoSQL and SQL databases.
- Purpose of comparison: understanding key differences and use cases.

Slide 2: Overview of MongoDB (NoSQL)

- MongoDB (NoSQL) Overview :
- Data Model: Document-oriented, stores data in JSON-like BSON format.
- Schema: Flexible schema design, allows dynamic and unstructured data.
- Scalability: Horizontal scaling (sharding) for handling large volumes of data.
- Use Cases: Big data applications, real-time analytics, content management.

Slide 3: Overview of SQL Databases

- SQL Databases Overview :
- Examples: MySQL, PostgreSQL, Oracle, SQL Server.
- Data Model: Table-based, stores data in rows and columns.
- Schema: Rigid schema design, requires predefined schema with fixed structure.
- Scalability: Vertical scaling (adding resources to a single server) and limited horizontal scaling.

Slide 4: Comparison

- MongoDB vs. SQL Databases Comparison :
- Data Structure:
 - MongoDB: Schema-less, document-based.
 - SQL: Structured, table-based with predefined schema.
- Scalability:
 - MongoDB: Horizontal scaling with sharding.
 - SQL: Vertical scaling, with some support for horizontal scaling.
- Query Language:
 - MongoDB: Uses JSON-like query language.
 - SQL: Uses Structured Query Language (SQL).

- **Transactions:**
 - MongoDB: Supports multi-document transactions, but historically weaker in ACID compliance.
 - SQL: Strong ACID (Atomicity, Consistency, Isolation, Durability) compliance.
- **Use Cases:**
 - MongoDB: Suitable for applications with evolving data models, unstructured data.
 - SQL: Best for applications requiring complex queries, strong consistency, and structured data.

Slide 5: Conclusion

- **Title:** Summary & Recommendations
- **Content:**
 - **Summary:** Recap of key differences and strengths of MongoDB and SQL databases.
 - **Recommendations:**
 - Choose MongoDB for projects needing flexibility, scalability, and handling of diverse data types.
 - Choose SQL databases for projects requiring strong data integrity, complex queries, and structured data.
 - **Considerations:** Evaluate based on project requirements, data structure, scalability needs, and transaction requirements.