MongoDB vs SQL

Introduction to Databases Checkpoint

Introduction to SQL and NoSQL

SQL Databases

- Structured Query Language (SQL) used for defining and manipulating data
- Relational Databases (e.g., MySQL, PostgreSQL)
- Data is stored in tables with rows and columns

MongoDB (NoSQL Database)

- Document-oriented NoSQL database
- Stores data in flexible, JSON-like documents
- Suitable for unstructured or semi-structured data

Data Structure and Flexibility

Feature	SQL Database	MongoDB (NoSQL)
Relationships	Uses joins and foreign keys	Embedded documents instead of joins
Use Case	Structured data	Unstructured/semi-structured data
Schema	Fixed schema	Dynamic schema (flexible)
Structure	Tables (rows & columns)	Collections (JSON documents)

Performance and Scalability

SQL Databases:

- **Vertical scaling**: Scale by upgrading hardware (CPU, RAM)
- Good for complex queries involving multiple tables
- Slower with high volume writes

MongoDB:

- **Horizontal scaling**: Scale by adding more servers
- Faster with large-scale, write-heavy applications
- Ideal for real-time data and IoT

Query Language and Transactions

Feature	SQL Database	MongoDB (NoSQL)
Query Language	SQL (standardized)	Query API with JSON-like syntax
Transactions	Strong ACID compliance	Supports ACID transactions (limited scope)
Complex Queries	Good for multi-table joins	Not optimized for joins; prefers embedding data

When to Use Each

Use SQL Databases When:

- Data is highly structured
- You need complex joins and relationships
- ACID compliance is critical

Use MongoDB When:

- Flexibility in data structure is required
- Working with large datasets and unstructured data
- Need for horizontal scaling and high-speed data ingestion