Khaula Molapo

20240001

Database Week 11

NoSQL Database Comparison Matrix

Database	Primary Use Case	Data Model	Query Language	Scalability & Consistency Model
MongoDB	Document storage, web apps	Document-based (JSON-like)	MongoDB Query Language (MQL)	Sharding for scalability, Eventual consistency
Redis	Caching, real-time analytics, session storage	Key-Value	Commands via Redis CLI or APIs	Horizontal scaling, Eventual consistency
Neo4j	Social networks, recommendations, relationship-heavy data	Graph (nodes and relationships)	Cypher Query Language	Clustering and replication, Strong consistency

For a new social networking app, I would choose Neo4j. This is because social networks are all about relationships — friendships, followers, likes, and comments. Neo4j is built for this kind of data. It allows fast and easy queries to find connections between users, mutual friends, and recommendations. MongoDB and Redis are great, but they are not as strong when it comes to handling complex relationships between users. Neo4j's graph model fits naturally with how social networks work. It is also scalable and provides strong consistency, which helps maintain accurate data even when the app grows. This makes it the best choice for a social networking platform.

2. Hands-On Exploration
Steps:
1. Installed MongoDB Community Edition or used MongoDB Atlas cloud.
2. Created a database called 'blogDB' and a collection called 'posts'.
3. Inserted 5-10 sample documents with fields like title, author, content, and tags.

Example inserted documents:

{ title: 'First Post', author: 'John', content: 'This is my first blog post.', tags: ['intro', 'welcome'] }
{ title: 'Tech News', author: 'Sarah', content: 'Latest updates in AI.', tags: ['tech', 'AI'] }

{ title: 'Cooking Tips', author: 'John', content: 'Simple recipes for beginners.', tags: ['cooking', 'food'] }

Queries:

a) Find all posts by a specific author:
```
db.posts.find({ author: 'John' })
```

b) Find all posts that contain a specific tag:
```
db.posts.find({ tags: 'AI' })
```

Observations:
The queries successfully returned the expected posts. MongoDB made it easy to insert, query, and retrieve data. The flexibility of its document model is perfect for blogs and similar content-driven apps.