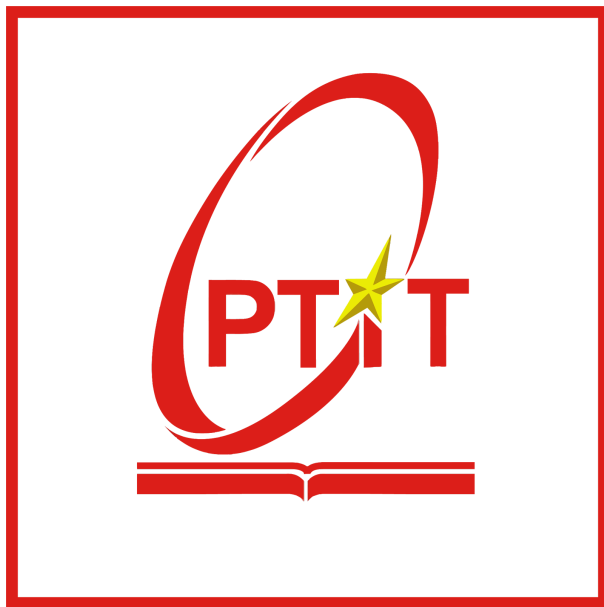


**BỘ THÔNG TIN VÀ TRUYỀN THÔNG  
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

---



**BÁO CÁO THỰC TẬP CƠ SỞ TUẦN 4**

**Tổng quan kiến thức cơ bản về  
MongoDB**

Giảng viên hướng dẫn: TS. Kim Ngọc Bách

Sinh viên thực hiện:

- Nguyễn Quang Trung - B22DCDT321

# MỤC LỤC

<b>Cơ sở Dữ liệu NoSQL.....</b>	<b>4</b>
Lịch sử phát triển của NoSQL.....	4
Một số thuật ngữ trong NoSQL.....	5
<b>MongoDB và MySQL khác nhau ở điểm nào?.....</b>	<b>6</b>
Những điểm tương đồng giữa MongoDB và MySQL là gì?.....	6
Giấy phép nguồn mở.....	6
Hỗ trợ lập chỉ mục.....	6
Giao diện thân thiện với người dùng.....	7
Ngôn ngữ lập trình.....	7
Bảo mật.....	7
Tài liệu và hỗ trợ từ cộng đồng.....	7
Điểm khác biệt chính giữa MongoDB và MySQL.....	7
Mô hình dữ liệu.....	8
Khả năng điều chỉnh quy mô.....	8
Hiệu năng.....	9
Sự linh hoạt.....	9
Kiểm soát truy cập.....	10
So sánh về thời điểm nên sử dụng MongoDB hay MySQL.....	10

## Cơ sở Dữ liệu NoSQL

Cơ sở dữ liệu NoSQL (viết tắt của “Not SQL” hoặc “Not Only SQL”) là một cơ sở dữ liệu không quan hệ (Non-Relational). NoSQL là tên gọi chung cho các cơ sở dữ liệu không sử dụng các bảng như MySQL hay SQL Server. Nó không sử dụng mô hình cơ sở dữ liệu quan hệ (RDBMSs) để lưu trữ và truy cập dữ liệu mà sử dụng mô hình lưu trữ key - value. Thay vì lưu trữ dữ liệu trong các bảng như một cơ sở dữ liệu truyền thống, một CSDL NoSQL lưu trữ từng hạng mục riêng với một khóa duy nhất. Ngoài ra, NoSQL còn là một hệ thống sử dụng dữ liệu phân tán. Tức là dữ liệu của hệ thống này không chỉ bao gồm những gì mà chúng ta nhập vào mà còn có thể là dữ liệu từ các file hoặc video,... và chúng được lưu trữ ở nhiều server khác nhau. Điều này cung cấp một cách tiếp cận nhiều linh hoạt hơn để lưu trữ dữ liệu hơn là một cơ sở dữ liệu quan hệ.

Với việc lưu trữ dữ liệu phân tán, NoSQL hỗ trợ rất tốt trong việc xử lý một khối lượng lớn dữ liệu lên đến hàng petabytes mà không cần phải có tài nguyên phần cứng cao. Nó chia dữ liệu của mình ra để lưu trữ trên nhiều Server khác nhau tạo điều kiện cho việc mở rộng theo chiều ngang một cách thuận lợi với chi phí không quá cao. Bên cạnh đó, việc lưu trữ trên nhiều Server cũng làm cho NoSQL có độ chịu tải và độ chịu lỗi cao. Nhiều hệ thống cơ sở dữ liệu NoSQL cũng sử dụng mã nguồn mở.

## Lịch sử phát triển của NoSQL

- Thuật ngữ NoSQL được giới thiệu lần đầu tiên bởi Carlo Stozzi vào năm 1998, lúc này nó còn được gọi là *Lightweight open source relational database*. Tuy nhiên, các khái niệm về cơ sở dữ liệu này vẫn chưa được hoàn thiện và vẫn còn mang hướng cơ sở dữ liệu quan hệ. Đến đầu năm 2009, trong sự kiện thảo luận về “các cơ sở dữ liệu phân tán, không quan hệ nguồn mở”, Johan Oskarsson của Last.fm đã giới thiệu lại thuật ngữ NoSQL.
- Đây là một cột mốc đánh dấu sự phát triển của hệ cơ sở dữ liệu mới - NoSQL. Từ đó, NoSQL trở nên phát triển và phổ biến và đặc biệt thông dụng với các mạng xã hội như: Facebook, Twitter,...

## **Một số thuật ngữ trong NoSQL**

- Ràng buộc (Relational)
- Không ràng buộc (Non Relational)
- Khả năng mở rộng (High Scalability)
- Khả năng mở rộng theo chiều dọc (Vertical scalable)
- Khả năng mở rộng theo chiều ngang (Horizontal scalable)
- Phân tán dữ liệu (Distributed data)
- Triển khai linh hoạt (Deployment Flexibility)
- Tính sẵn sàng (High Availability)
- Nhất quán cuối (Eventual consistency)
- Lưu trữ tốt (Durability)

# MongoDB và MySQL khác nhau ở điểm nào?

MongoDB và MySQL là hai hệ thống quản lý cơ sở dữ liệu mà bạn có thể sử dụng để lưu trữ và quản lý dữ liệu. MySQL là một hệ thống cơ sở dữ liệu quan hệ lưu trữ dữ liệu ở định dạng bảng có cấu trúc. Ngược lại, MongoDB lưu trữ dữ liệu dưới dạng tài liệu JSON ở định dạng linh hoạt hơn. Cả hai hệ thống này đều mang lại hiệu năng và khả năng điều chỉnh quy mô, nhưng mỗi hệ thống sẽ có hiệu năng cao hơn tùy vào các trường hợp sử dụng khác nhau.

## Những điểm tương đồng giữa MongoDB và MySQL là gì?

MySQL và MongoDB đều là hệ thống quản lý cơ sở dữ liệu. Chúng lưu trữ dữ liệu và sở hữu giao diện người dùng cũng như ngôn ngữ truy vấn tích hợp sẵn, vì vậy bạn có thể thêm, chỉnh sửa, sửa đổi và phân tích dữ liệu.

## Giấy phép nguồn mở

MySQL và các phiên bản ban đầu của MongoDB đều có giấy phép nguồn mở. Bạn có thể tải xuống miễn phí các phiên bản nguồn mở. Sau đó, bạn có thể sửa đổi mã tùy theo nhu cầu của mình.

MySQL tuân thủ theo Giấy phép công cộng GNU. Tất cả các phiên bản của MongoDB được phát hành trước ngày 16 tháng 10 năm 2018 đều có sẵn theo Giấy phép công cộng GNU Affero.

## Hỗ trợ lập chỉ mục

MySQL và MongoDB sử dụng lập chỉ mục để cải thiện tốc độ truy vấn và hiệu năng. Chỉ mục là cấu trúc cơ sở dữ liệu liên kết với dữ liệu được truy cập thường xuyên. Chỉ mục giúp tìm kiếm và truy xuất dữ liệu nhanh chóng.

Cả hai nền tảng cơ sở dữ liệu MySQL và MongoDB đều sử dụng chỉ mục băm, chỉ mục B-tree cùng nhiều loại chỉ mục khác.

## **Giao diện thân thiện với người dùng**

MongoDB và MySQL đều rất dễ sử dụng. Chúng dùng ngôn ngữ truy vấn dựa trên ngôn ngữ tự nhiên để cập nhật và đọc dữ liệu. Chúng cũng cung cấp một giao diện người dùng đồ họa (GUI) để quản lý và phân tích dữ liệu một cách trực quan hơn.

## **Ngôn ngữ lập trình**

MySQL và MongoDB cùng tương thích với nhiều ngôn ngữ lập trình giống nhau. Bạn có thể sử dụng Java, Python, Node.js, PHP phía máy chủ, Ruby và C# với cả MongoDB và MySQL.

## **Bảo mật**

MySQL và MongoDB đều sử dụng xác thực, kiểm soát truy cập và mã hóa để đảm bảo tính bảo mật cho cơ sở dữ liệu. Chúng sử dụng mã hóa TLS/SSL để bảo vệ dữ liệu đang được truyền và đang được lưu trữ. Chúng cũng cho phép bạn xác định các cấp độ khác nhau cho quyền truy cập của người dùng.

## **Tài liệu và hỗ trợ từ cộng đồng**

MySQL và MongoDB đều có tài liệu chi tiết chính thức trên trang web. Các video và tài liệu hướng dẫn của mỗi hệ thống chứa đầy đủ các chỉ dẫn để cài đặt, định cấu hình và chạy các tác vụ hoạt động.

MongoDB và MySQL cũng có một cộng đồng các nhà phát triển tích cực giải đáp những thắc mắc và giúp bạn khắc phục sự cố. Cả hai hệ thống đều cung cấp các phiên bản doanh nghiệp với sự hỗ trợ chuyên sâu cho các yêu cầu cụ thể của bạn.

## **Điểm khác biệt chính giữa MongoDB và MySQL**

MySQL là một hệ thống quản lý cơ sở dữ liệu quan hệ, còn MongoDB là một hệ thống cơ sở dữ liệu NoSQL.

MySQL sử dụng SQL, thứ mà hầu hết các nhà phát triển đều có kinh nghiệm sử dụng. Ngược lại, MongoDB sử dụng Ngôn ngữ truy vấn MongoDB (MQL).

Mặc dù MQL và SQL có những điểm tương đồng, MQL thường đòi hỏi bỏ nhiều công sức tìm hiểu hơn.

Tiếp theo, chúng tôi sẽ đề cập đến một số điểm khác biệt chính khác.

### **Mô hình dữ liệu**

MySQL là một hệ thống cơ sở dữ liệu quan hệ lưu trữ dữ liệu trong cột, hàng và bảng. Bạn lưu trữ dữ liệu trong hàng, với mỗi cột đại diện cho một loại dữ liệu khác nhau. Sau đó, bạn xác định mối quan hệ giữa dữ liệu bằng cách sử dụng khóa ngoại và khóa chính. Mỗi bảng có một khóa chính mà bạn sử dụng để xác định dữ liệu, còn khóa ngoại tạo ra mối quan hệ.

MongoDB là một cơ sở dữ liệu chuyên tài liệu lưu trữ tất cả dữ liệu dưới dạng các tài liệu JSON nhị phân (BSON). BSON cho phép bạn tuần tự hóa nhiều dạng dữ liệu. Sử dụng tài liệu BSON cho phép bạn lưu trữ dữ liệu phi cấu trúc, bán cấu trúc và có cấu trúc. Thay vì lược đồ cơ sở dữ liệu, MongoDB sử dụng cách tiếp cận linh hoạt, lưu trữ tài liệu trong các tập hợp.

### **Khả năng điều chỉnh quy mô**

Trong một hệ thống cơ sở dữ liệu MySQL, chỉ có một số tùy chọn có sẵn giới hạn để mở rộng quy mô. Bạn có thể chọn từ các tùy chọn sau:

- Khả năng điều chỉnh quy mô theo chiều dọc bằng cách thêm nhiều tài nguyên vào máy chủ cơ sở dữ liệu hiện tại
- Sao chép chỉ có quyền đọc bằng cách tạo bản sao chỉ đọc của cơ sở dữ liệu trên các máy chủ khác

Sao chép chỉ có quyền đọc bị giới hạn tối đa năm bản sao. Các bản sao cũng có thể không cập nhật kịp bản sao chính, điều này tạo ra các vấn đề về hiệu năng trên quy mô lớn. Khả năng điều chỉnh quy mô theo chiều dọc cũng bị giới hạn trong cơ sở hạ tầng mà bạn sử dụng.

Ngược lại, thiết kế MongoDB mang lại lợi thế đáng kể về khả năng điều chỉnh quy mô. MongoDB có hai tính năng chính để điều chỉnh quy mô:

- Tập bản sao – nhóm máy chủ MongoDB chứa dữ liệu giống hệt nhau
- Phân mảnh – các phần khác nhau của dữ liệu được phân tán trên các máy chủ khác nhau

MongoDB cho phép bạn tạo các cụm được phân mảnh, nhờ đó các phần của dữ liệu của bạn được sao chép trên nhiều máy chủ. Ví dụ: nếu bạn có một số lượng lớn hồ sơ khách hàng, bạn có thể phân tán chúng để các hồ sơ có tên từ A-J và các hồ sơ có tên từ K-Z nằm trong các tập bản sao riêng. MongoDB có thể điều chỉnh quy mô theo chiều ngang để tối ưu hóa cả hiệu năng đọc và ghi trên quy mô lớn.

### **Hiệu năng**

MySQL được thiết kế để thực hiện các phép ghép nối hiệu năng cao trên nhiều bảng được lập chỉ mục thích hợp. Tuy nhiên, MySQL yêu cầu dữ liệu phải được chèn theo từng hàng, vì vậy hiệu năng ghi sẽ chậm hơn.

Các tài liệu MongoDB tuân theo một mô hình dữ liệu phân cấp và lưu giữ hầu hết dữ liệu trong một tài liệu duy nhất, giảm nhu cầu ghép nối trên nhiều tài liệu. Các phép ghép nối được hỗ trợ thông qua thao tác \$lookup, nhưng chúng không được tối ưu hóa cho hiệu năng. Tuy nhiên, MongoDB cung cấp một API insertMany() để nhanh chóng chèn dữ liệu, ưu tiên hiệu năng ghi.

### **Sự linh hoạt**

Là một hệ thống quản lý cơ sở dữ liệu quan hệ, MySQL có cấu trúc cứng nhắc hơn MongoDB. MySQL sử dụng một lược đồ cố định và sắp xếp dữ liệu thành hàng và bảng. Bạn phải tạo cấu trúc dữ liệu và đưa dữ liệu vào một hệ thống dạng bảng để sử dụng MySQL.

Bằng cách lưu trữ dữ liệu dưới dạng tài liệu JSON, MongoDB cho phép bạn xây dựng các ứng dụng phức tạp với nhiều kiểu dữ liệu riêng biệt. Ví dụ: bạn có thể



tạo các trường mới bằng cách cập nhật những trường mảng lồng nhau. Bạn cũng có thể sử dụng quy trình tổng hợp – một tính năng của MongoDB cho phép bạn chuyển đổi dữ liệu bằng cách kết hợp nhiều hoạt động thành một quy trình làm việc duy nhất.

### **Kiểm soát truy cập**

Trong MongoDB, bạn có thể kiểm soát truy cập ở cấp độ hoạt động, tập hợp hoặc cơ sở dữ liệu. Hệ thống này sử dụng chứng chỉ Kerberos, X.509 và LDAP để xác thực người dùng. Ngược lại, MySQL cho phép bạn hạn chế quyền truy cập của người dùng ở cấp độ người dùng, cơ sở dữ liệu và bảng. MySQL sử dụng hệ thống xác thực riêng. Hệ thống này tạo ra thêm một lỗ hổng bảo mật trong các cuộc tấn công đưa SQL vào, điều mà phương thức tiếp cận không lược đồ của MongoDB tránh được.

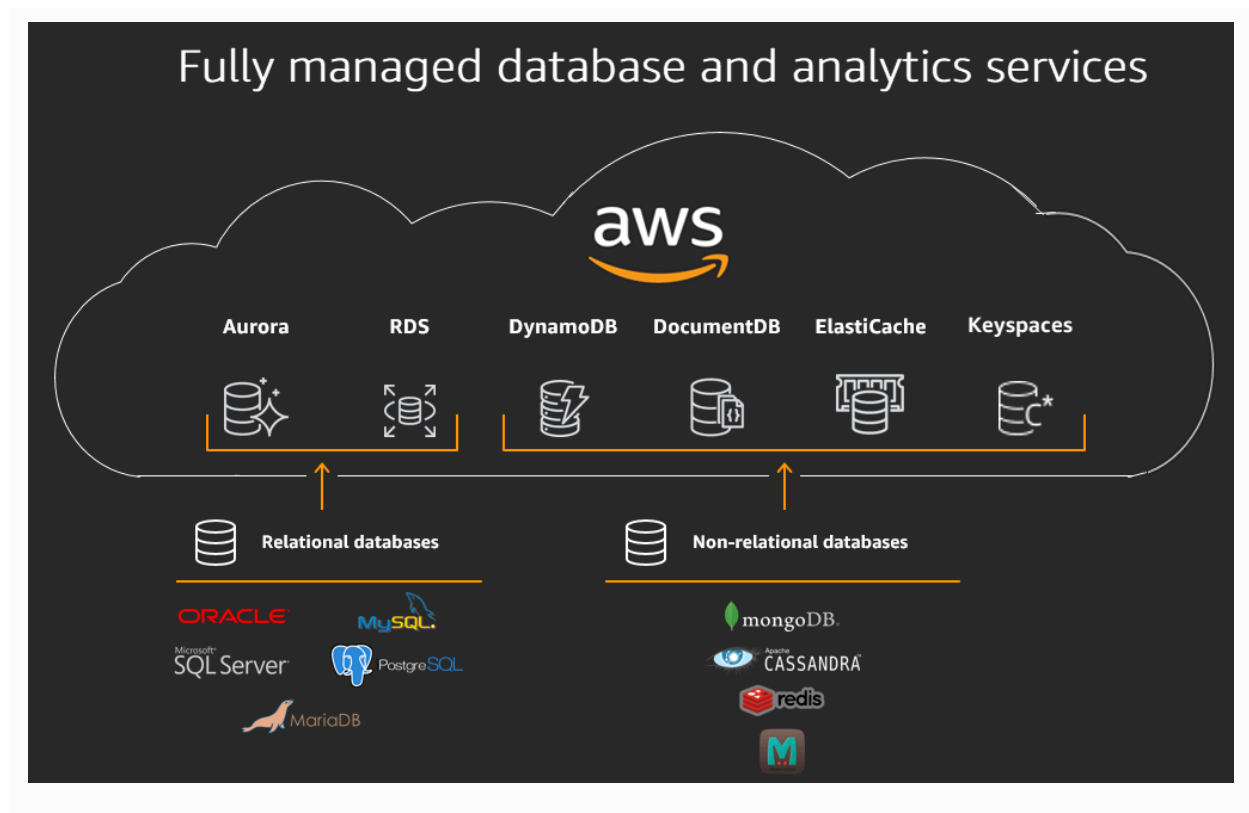
### **So sánh về thời điểm nên sử dụng MongoDB hay MySQL**

Định dạng kho lưu trữ dữ liệu trong MySQL khiến MySQL phù hợp với việc lưu kho dữ liệu và xử lý phân tích trực tuyến. MySQL tuân thủ ACID, tức là các giao dịch có tính nguyên tử, nhất quán, cô lập và lâu bền. Điều này khiến MySQL có ích khi bạn làm việc với các giao dịch phức tạp, chẳng hạn như trong các trường hợp sử dụng thương mại điện tử, giao dịch và tài chính.

Các dữ liệu có cấu trúc chặt chẽ và lập chỉ mục của MySQL cũng khiến MySQL phù hợp với cho các truy vấn đặc thù. Truy vấn đặc thù thường được thực hiện bởi người dùng cuối hoặc các nhà phân tích dữ liệu cần phải truy cập nhanh vào dữ liệu không có sẵn qua báo cáo hoặc truy vấn định sẵn.

Mặt khác, MongoDB là cơ sở dữ liệu NoSQL. MongoDB sẽ thích hợp hơn trong trường hợp bạn làm việc với dữ liệu phi cấu trúc trong các trường hợp sử dụng như mạng xã hội, phương tiện truyền thông hoặc Internet vạn vật (IoT). Vì

MongoDB không có lược đồ, nên đây là lựa chọn tốt để xử lý dữ liệu thay đổi và mở rộng liên tục.



	MongoDB	MySQL
Mô hình dữ liệu	MongoDB lưu trữ dữ liệu trong các tài liệu JSON, sau đó sắp xếp chúng thành các tập hợp.	MySQL lưu trữ dữ liệu trong các cột và hàng. Kho lưu trữ dữ liệu có dạng bảng và có tính quan hệ.
Khả năng mở rộng	MongoDB sử dụng phương pháp sao chép và phân mảnh để	MySQL sử dụng khả năng điều chỉnh quy mô theo chiều dọc và

	điều chỉnh quy mô theo chiều ngang.	bản sao chỉ có quyền đọc để cải thiện hiệu suất ở quy mô lớn.
Ngôn ngữ truy vấn	MongoDB sử dụng Ngôn ngữ truy vấn MongoDB.	MySQL sử dụng ngôn ngữ truy vấn SQL.
Hiệu năng	MongoDB vượt trội trong việc chèn hoặc cập nhật một số lượng lớn bản ghi.	Tốc độ MySQL nhanh hơn khi chọn một số lượng lớn bản ghi.
Tính linh hoạt	MongoDB không có lược đồ, nhờ đó mang lại tính linh hoạt cao hơn và cho phép cơ sở dữ liệu này thao tác với dữ liệu phi cấu trúc, bán cấu trúc và có cấu trúc.	MySQL có một lược đồ cứng nhắc rất phù hợp với dữ liệu có cấu trúc.
Bảo mật	MongoDB sử dụng các chứng chỉ Kerberos, X.509 và LDAP để xác thực người dùng.	MySQL sử dụng phương pháp xác thực tích hợp sẵn.

## Những câu lệnh cơ bản của MongoDB

Sự tương quan giữa các thuật ngữ được sử dụng trong RDBMS và MongoDB

RDBMS	MongoDB
Database	Database
Table	Collection
Tuple/Row	Document
column	Field
Table Join	Embedded Documents
Primary Key	Primary Key(mặc định là id)

### Một số câu lệnh được sử dụng trong MongoDB

#### 1. Tạo database trong MongoDB

```
use DatabaseName
```

Trong đó DatabaseName là tên của database các bạn muốn tạo. vd: Tạo database có tên là projectdemo.

```
use projectdemo
```

#### 2. Xem database đang sử dụng

Để xem database đang sử dụng(current database) thì chúng ta sử dụng lệnh:

```
db
```

### 3. Xem tất cả các database trong hệ thống

Để xem tất cả các database đã được tạo trên MongoDB thì mọi người sử dụng lệnh sau:

```
show dbs
```

Chú ý: Lệnh này sẽ chỉ hiện ra các database đã có ít nhất một collection(hiểu như table trong MySql), còn nếu chưa có thì nó sẽ không hiện ra.

### 4. Lấy tất cả dữ liệu trong collection

Để lấy tất cả dữ liệu trong collection chúng ta sử dụng phương thức find() với cú pháp sau:

```
db.collectionName.find()
```

Trong đó collectionName là tên của collection mà các bạn muốn truy vấn. Nếu như bạn muốn dữ liệu trả về được hiển thị theo cấu trúc đã được định sẵn thì chỉ cần thêm hàm pretty() vào phía sau hàm find().

```
db.collectionName.find().pretty()
```

### 5. Truy vấn có điều kiện trong MongoDB

Để truy vấn có điều kiện trong MongoDB thì bạn cũng sử dụng cú pháp tương tự như phần 1, nhưng lúc này chúng ta sẽ thêm điều kiện vào trong hàm find() với cú pháp sau:

```
db.collectionName.find(condition)
```

**Cú pháp của các mệnh đề điều kiện:**

Phép toán	Cú pháp
Bằng(Equality)	{key: value}
Nhỏ hơn (Less Than)	{key: {\$lt: value}}
Nhỏ hơn bằng (Less Than Equals)	{key: {\$lte: value}}
Lớn hơn (Greater Than)	{key: {\$gt: value}}
Lớn hơn bằng (Greater Than Equals)	{key: {\$gte: value}}
Khác (Not Equals)	{key: {\$ne: value}}
Trong ( In)	{key: {\$in: [value1, value2,..]}}
Không Thuộc (Not In)	{key: {\$nin: [value1, value2,..]}}