



Lập trình ứng dụng web bằng NodeJS

Bài 8. *Xây dựng web với CSDL MongoDB*

<http://csc.edu.vn/>





Nội dung

1. MongoDB là gì?
2. Khi nào nên sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



1. MongoDB là gì?

- ❑ Hiểu một cách nôm na thì **MongoDB** là một mã nguồn mở và là một tập tài liệu dùng cơ chế NoSQL để truy vấn, nó được viết bởi ngôn ngữ C++. Chính vì được viết bởi C++ nên nó có khả năng tính toán với tốc độ cao chứ không giống như các hệ quản trị CSDL hiện nay.
- ❑ MongoDB lưu trữ dữ liệu với cấu trúc JSON
- ❑ Hoạt động dựa trên khái niệm document và collection



1. MongoDB là gì?

- ❑ **Collection:** Collection trong MongoDB là nhóm các tài liệu (document), nó tương đương với một bảng (table) trong CSDL thông thường nên mỗi collection sẽ thuộc về một database duy nhất.
- ❑ Tuy nhiên nó có một sự khác biệt đó là nó không có ràng buộc Relationship như các hệ quản trị CSDL khác nên việc truy xuất rất nhanh, chính vì thế mỗi collection có thể chứa nhiều thể loại khác nhau không giống như table trong hệ quản trị mysql là các field cố định



1. MongoDB là gì?

- ❑ **Document:** Document trong MongoDB có cấu trúc tương tự như kiểu dữ liệu JSON, nghĩa là sẽ có các cặp (key => giá trị) nên nó có tính năng động rất lớn.
- ❑ Document ta có thể hiểu nó giống như các record dữ liệu trong MYSQL, tuy nhiên nó có sự khác biệt là các cặp (key => value) có thể không giống nhau ở mỗi document.



1. MongoDB là gì?

❑ Cấu trúc một dữ liệu trong document sẽ như sau:

```
{
  "_id" : ObjectId("5881ae7db9a27b4879e2411a"),
  "ho_ten" : "Hùng Nguyễn",
  "ngay_dang_ky" : "2016-12-5",
  "tai_khoan" : "hungnguyen",
  "mat_khau" : "123456",
  "email" : "hungnguyenxuan118@gmail.com",
  "gioi_tinh" : 0,
  "avatar" : "nam1.jpg",
  "id_quyen_han" : "58a2ca401d2c5cd581eadfcf",
  "online" : 1,
  "thoi_gian_dang_nhap" : 1488277349
}
```



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



2. Khi nào sử dụng MongoDB

- ❑ Nếu CSDL cần thêm dữ liệu với tốc độ nhanh và bảo mật.
- ❑ Xây dựng ứng dụng với CSDL cực lớn, ví dụ như cần khoảng 10 triệu dòng dữ liệu thì đó là “ác mộng” với các CSDL sử dụng SQL.
- ❑ Xây dựng ứng dụng với kỹ thuật thời gian thực nhiều. Vì trong quá trình nhiều người dùng cùng tương tác mà xảy ra lỗi thì MongoDB sẽ bỏ qua nó



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



3. Cài đặt CSDL MongoDB

- ❑ Download chương trình cài đặt MongoDB tại trang chủ của nó: <https://www.mongodb.com/download-center#community>

The screenshot shows the MongoDB download page for the Community Server. The page has a green header with navigation tabs: Community Server, Enterprise Server, Ops Manager, Compass, and Connector for BI. Below the header, there are links for Current Release, Previous Releases, and Development Releases. The main content area displays the 'Current Stable Release (3.4.4)' with a date of 04/20/2017 and links to Release Notes and Changelog. It also shows the download source as tgz or zip. A dropdown menu for 'Version' is set to 'Windows Server 2008 R2 64-bit and later, with SSL support x64'. Below this, there is a green button labeled 'DOWNLOAD (msi)'. At the bottom, there are links for 'Installation Instructions' and 'All Version Binaries'.



3. Cài đặt CSDL MongoDB

- ❑ Tạo thư mục để cài đặt CSDL của chúng ta vào, rồi sau đó chạy lệnh cài đặt:

```
C:\Program Files\MongoDB\Server\3.4\bin>mongod --dbpath=D:/data_mongo/
```

- ❑ Kết quả tạo thành công:

```
2016-09-04T07:26:01.700-0700 I CONTROL [main] ** WARNING: --rest is specified without --httpinterface,  
2016-09-04T07:26:01.701-0700 I CONTROL [main] ** enabling http interface
```



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



4. Áp dụng Robomongo để quản lý CSDL

- ❑ Download ứng dụng Robomongo tại website:
<https://robomongo.org/>

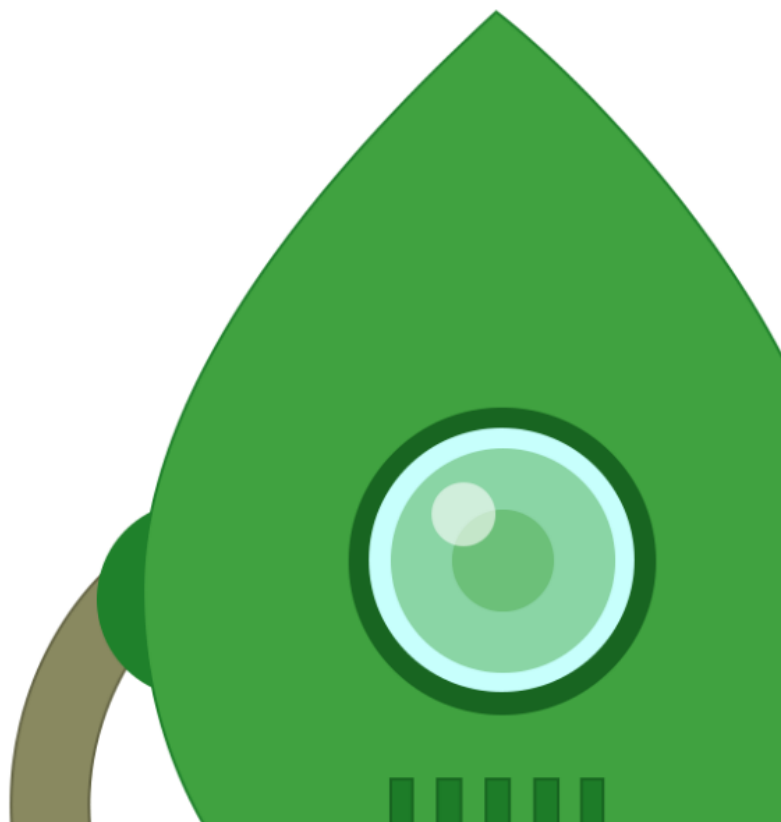
Robomongo & 3T

The Robomongo tool has been acquired by 3T Software Labs, the creators of Studio 3T (formerly MongoChef). 3T will continue development of Robomongo in parallel with its own MongoDB GUI and the good news is that it will be making Robomongo free for all users from now on. This is an exciting new step for Robomongo and its community, which will benefit users and the wider MongoDB ecosystem. [Click here to read Dmitry's blog explaining the reasons behind the deal](#) and check the [FAQ](#).



"I know the guys at 3T well and I know they'll do a great job of taking Robomongo forward — I can't wait to see the tool develop further through their backing and the support of the great Robomongo community!"

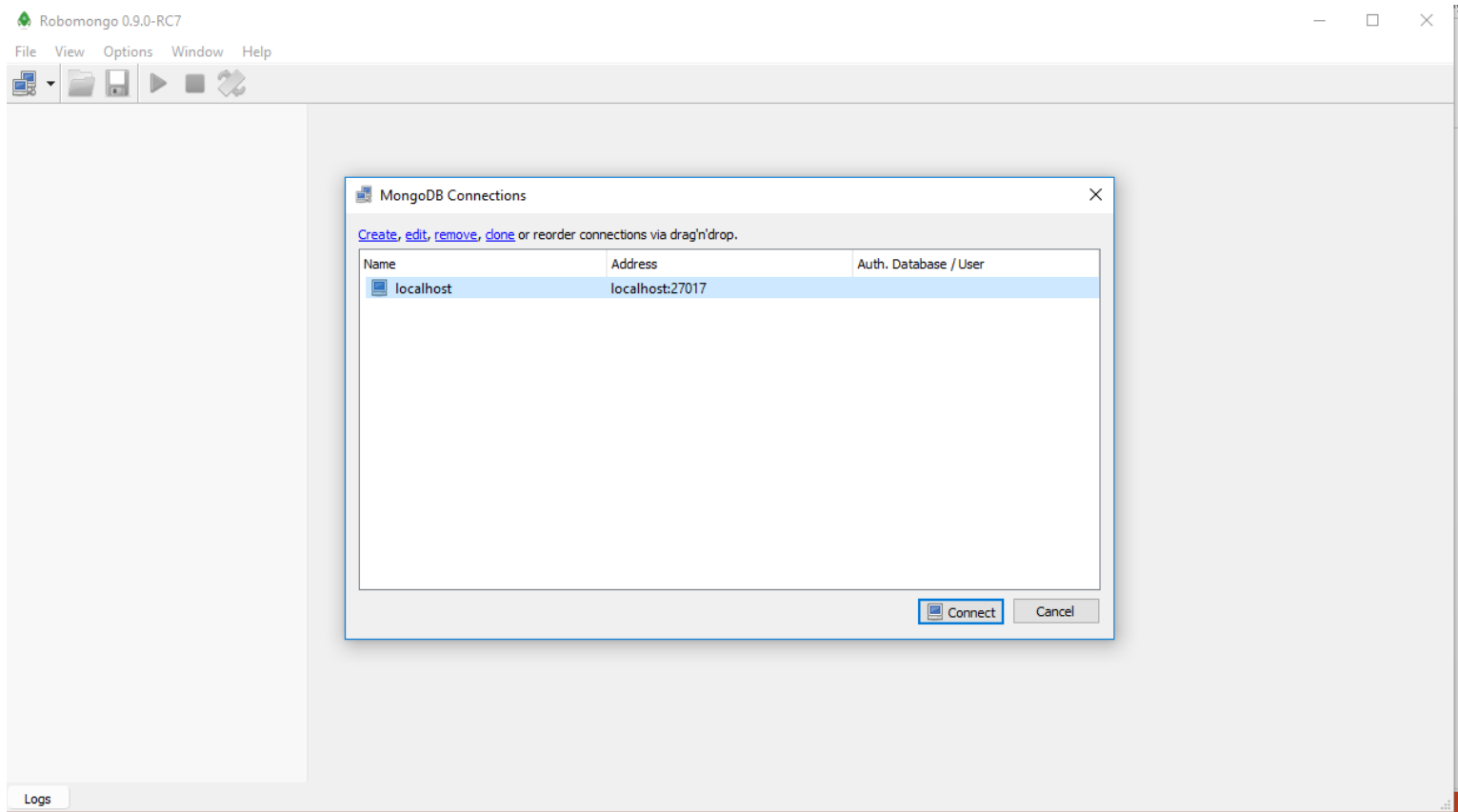
Download





4. Áp dụng Robomongo để quản lý CSDL

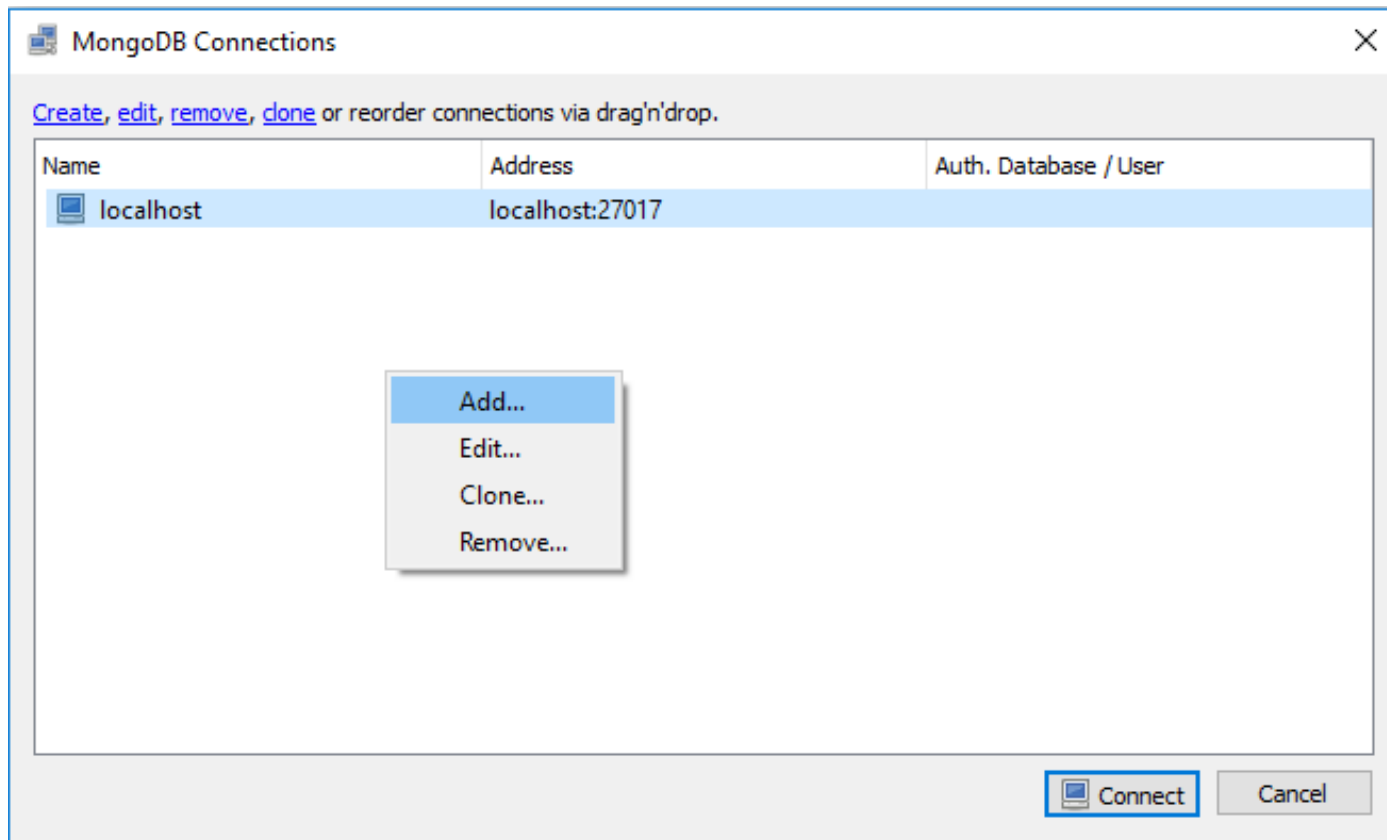
- ❑ Sau khi cài đặt xong khởi động chương trình sẽ có giao diện:





4. Áp dụng Robomongo để quản lý CSDL

❑ Tạo kết nối đến CSDL MongoDB cho Robomongo





4. Áp dụng Robomongo để quản lý CSDL

❑ Điền các thông tin cấu hình kết nối

The screenshot shows the 'Connection Settings' dialog box with three tabs: 'Connection', 'Authentication', and 'Advanced'. The 'Connection' tab is active. It contains the following fields and text:

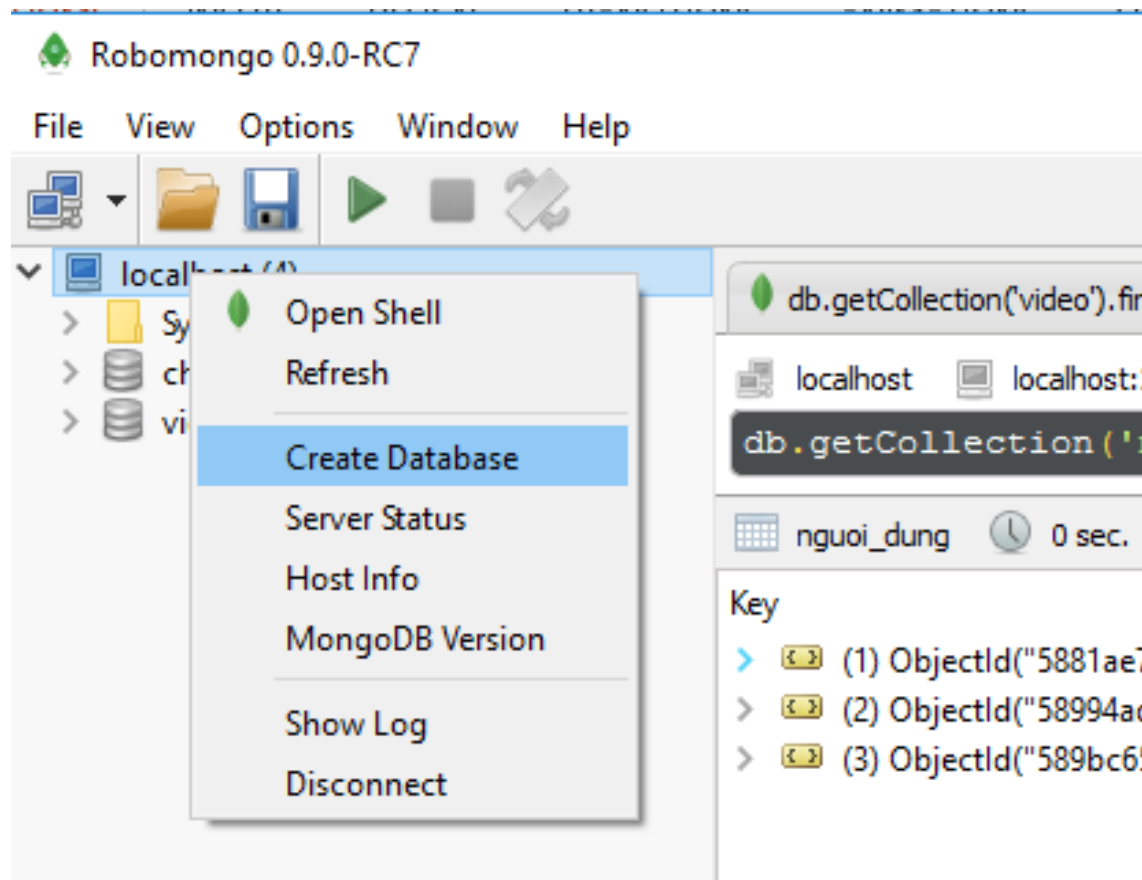
- Name:** A text box containing 'New Connection'. Below it, a note says: 'Choose any connection name that will help you to identify this connection.'
- Address:** Two text boxes. The first contains 'localhost' and the second contains '27017'. Below them, a note says: 'Specify host and port of MongoDB server. Host can be either IPv4, IPv6 or domain name.'

At the bottom of the dialog, there are three buttons: 'Test' (with an information icon), 'Save', and 'Cancel'.



4. Áp dụng Robomongo để quản lý CSDL

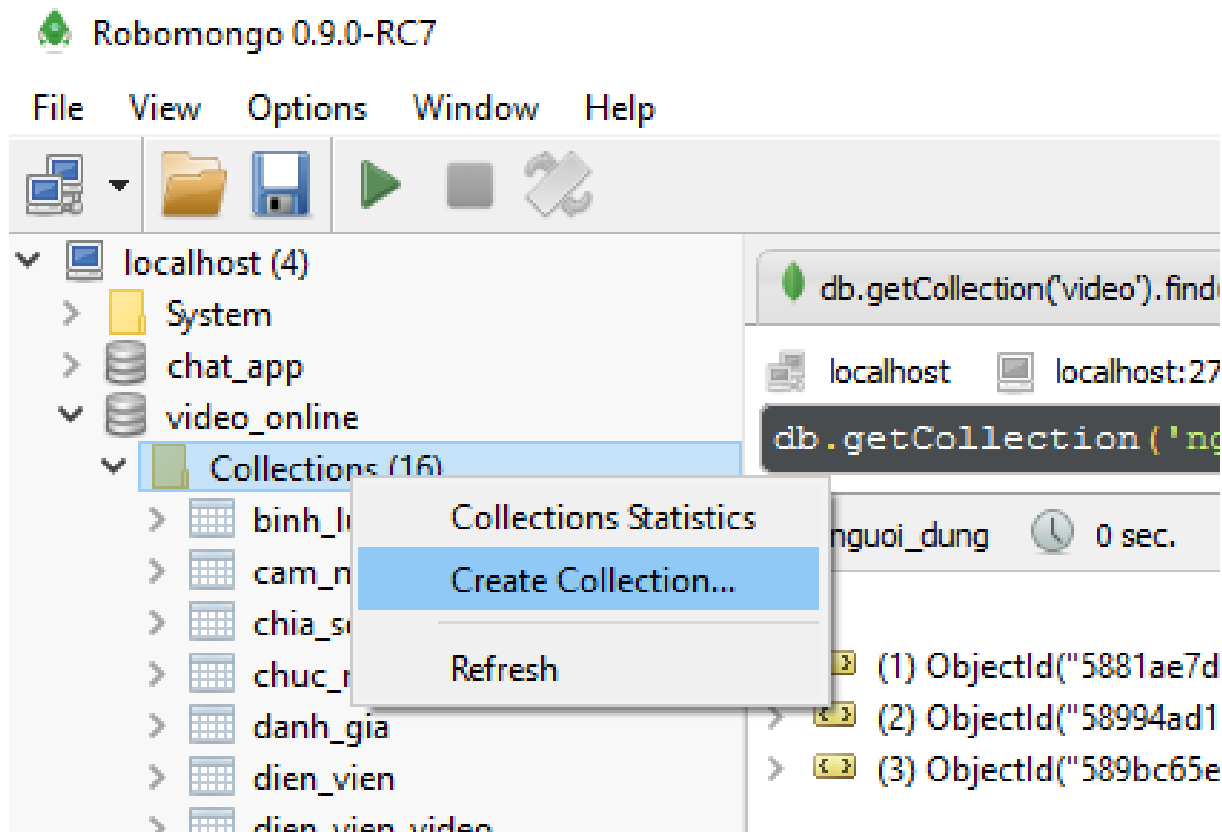
- ❑ Tạo Database MongoDB, nhấp chuột phải vào server sau khi kết nối và chọn như hình





4. Áp dụng Robomongo để quản lý CSDL

- ❑ Tạo mới một Collection cho Database, nhấp chuột phải vào Database đã tạo và chọn Create Collection





Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB

5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS



- ❑ Có nhiều thư viện khác nhau trong NodeJS hỗ trợ đọc CSDL từ MongoDB, ở đây chúng ta sẽ sử dụng thư viện MongoDB trong NodeJS để làm việc này.

- ❑ Cấu trúc lệnh để cài đặt từ terminal:

```
npm install mongodb --save
```

- ❑ Khi cài đặt thành công sẽ như hình sau:

```
-- mongodb@2.2.26
+-- mongodb-core@2.1.10
  -- readable-stream@2.2.7
    -- string_decoder@1.0.0
```



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. **Kết nối CSDL MongoDB**
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



6. Kết nối CSDL MongoDB

- ❑ Để kết nối đến CSDL Mongo, chúng ta sẽ chèn thư viện vào như câu lệnh sau:

```
var mongo_db = require("mongodb");
```

- ❑ Khởi tạo đối tượng MongoClient

```
var MongoClient = mongo_db.MongoClient;
```

- ❑ Khởi tạo đối tượng Database:

```
var db;
```



6. Kết nối CSDL MongoDB

❑ Cấu trúc câu lệnh kết nối:

```
MongoClient.connect("mongodb://localhost:27017/video  
_online", function(err, database){  
    if(err)  
    {  
        console.log(err);  
    }  
    db = database;  
});
```



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



7. Lấy danh sách dữ liệu MongoDB

- ❑ Đầu tiên để lấy thông tin từ CSDL MongoDB chúng ta cần chọn collection cần lấy ra với câu lệnh:

```
db.collection("menu", function(err, tên_biến_collection){  
    if(err)  
    {  
        //code xử lý khi bị lỗi  
    }  
    else  
    {  
        //code xử lý lấy dữ liệu collection thành công  
    }  
})
```



7. Lấy danh sách dữ liệu MongoDB

❑ Sau khi đã chọn Collection thành công, bắt đầu viết code để lấy dữ liệu ra sẽ có các lệnh cơ bản như sau:

- Lấy toàn bộ danh sách có trong collection
- Lấy duy nhất một dòng dữ liệu trong collection
- Lấy dữ liệu theo một điều kiện nhất định
- Lấy dữ liệu theo nhiều điều kiện và toán tử truy vấn
- Lấy danh sách có sắp xếp
- Lấy danh sách có giới hạn số dòng dữ liệu

7.1 Lấy toàn bộ danh sách có trong Collection



- ❑ Để lấy toàn bộ danh sách dữ liệu có trong Collection ta dùng cấu trúc lệnh như sau:

```
Tên_biến_collection.find({})
```

- ❑ Sau đó chuyển dữ liệu thành mảng đối tượng mới có thể sử dụng được thì dùng hàm `toArray()`, có cấu trúc:

```
Tên_biến_collection.find({}).toArray(function(err,  
tên_biến_danh_sách)
```

```
{
```

```
    //code xử lý khi đã lấy ra danh sách
```

```
}
```

7.2 Lấy duy nhất một dòng dữ liệu trong collection



- ❑ Đối với truy vấn chỉ lấy một dòng dữ liệu duy nhất, chúng ta sẽ dùng phương thức `findOne()`, với cấu trúc tương tự `find()`:

```
collection.findOne({tai_khoan: tai_khoan_post},  
function(err, nguoi_dung){  
    if(err)  
        console.log(err);  
    else  
    {  
        //code xử lý lấy dữ liệu collection thành công  
    }  
})
```



7.3 Lấy dữ liệu theo một điều kiện nhất định

- ❑ Để thực hiện truy vấn lấy dữ liệu theo điều kiện nhất định thì sẽ truyền tham số theo dạng đối tượng javascript vào trong hàm `find`, `findOne` với cấu trúc như sau:

```
findOne({tên_thuộc_tính: giá_trị})
```

- ❑ Ví dụ:

```
collection.findOne({tai_khoan: "hungnguyen"})
```

7.4 Lấy dữ liệu theo nhiều điều kiện và toán tử truy vấn



- ❑ Toán tử bé hơn: less than sẽ là ký tự “\$lt”, cách sử dụng như sau:

```
Biến_collection.find({ten_thuộc_tính: {$gt:  
giá_trị_so_sánh}})
```

- ❑ Tương tự cũng sẽ có toán tử lớn hơn, great than sẽ là ký tự “\$gt”, và có cách sử dụng như sau:

```
Biến_collection.find({ten_thuộc_tính: {$lt:  
giá_trị_so_sánh}})
```

7.4 Lấy dữ liệu theo nhiều điều kiện và toán tử truy vấn



- ❑ Toán tử and trong query mongodb sẽ là ký tự “\$and”, có phương pháp sử dụng:

```
{  
  $and: [{thuộc_tính_1: giá_trị}, {thuộc_tính_2:  
    giá_trị},...]  
}
```

- ❑ Toán tử or trong query mongodb là ký tự “\$or”, có cấu trúc:

```
{  
  $or: [{thuộc_tính_1 : giá_trị}, {thuộc_tính_2 :  
    giá_trị},...]  
}
```



7.5 Lấy danh sách có sắp xếp

❑ Chắc chắn rằng trong lúc truy vấn dữ liệu, chúng ta sẽ cần đến câu query có thể sắp xếp dữ liệu trả về. Và hiển nhiên trong MongoDB Query cũng có hỗ trợ việc sắp xếp với phương thức `sort()`.

❑ Có cấu trúc lệnh sử dụng như sau:

```
.sort({thuộc_tính_sắp_xếp: giá_trị_sắp_xếp})
```

❑ Lưu ý: giá trị sắp xếp ở đây là tăng dần hoặc giảm dần nhưng với giá trị số hiểu như sau: giá trị bằng 1 là tăng dần, giá trị bằng -1 là giảm dần

7.6 Lấy danh sách có giới hạn số dòng dữ liệu



- ❑ Nếu như làm trong MySQL để giới hạn dòng dữ liệu và bỏ qua các dòng dữ liệu sẽ có mệnh đề **LIMIT** để thực thi. Trong MongoDB sẽ có hàm **limit()**, **skip()** để thực thi nhiệm vụ giống như mệnh đề **LIMIT**.

- ❑ Cấu trúc hàm:

`limit(số_lượng_dòng_lấy_ra)`

`skip(số_lượng_dòng_bỏ_qua)`



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



8. Thêm dữ liệu MongoDB

- ❑ Thêm dữ liệu trong MongoDB chúng ta có hàm `insert()`. Và có cách sử dụng như sau:

```
Biến_collection.insert(thông_tin_thêm_vào_CSDL,
function(err, result){
    if(err){
        console.log(err);
    }
    else{
        //code xử lý sau khi thêm thành
        công
    }
});
```



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



9. Cập nhật dữ liệu MongoDB

- ❑ Dùng lệnh update của đối tượng collection trong NodeJS để cập nhật dữ liệu của CSDL mongoDB, với cấu trúc như sau:

```
Biến_collection.update({tên_thuộc_tính_tìm:  
giá_trị}, thông_tin_đối_tượng_lưu, function(err,  
result){  
    if(err){  
        console.log(err)  
    }  
    else{  
        //code xử lý sau khi cập nhật thành  
        công  
    }  
});
```



Nội dung

1. MongoDB là gì?
2. Khi nào sử dụng MongoDB
3. Cài đặt CSDL MongoDB
4. Áp dụng Robomongo để quản lý CSDL
5. Cài đặt thư viện xử lý CSDL MongoDB trong NodeJS
6. Kết nối CSDL MongoDB
7. Lấy danh sách dữ liệu MongoDB
8. Thêm dữ liệu MongoDB
9. Cập nhật dữ liệu MongoDB
10. Xóa dữ liệu MongoDB



10. Xóa dữ liệu MongoDB

- ❑ Để xóa dữ liệu trong MongoDB, chúng ta sẽ sử dụng phương thức `remove()` của đối tượng `collection`. Cấu trúc lệnh:

```
Biến_collection.remove({thuộc_tính_xác_định_document
_xóa: giá_trị}, function(err, result) {
    if (err) {
        console.log(err);
    }
    else
    {
        //code xử lý sau khi xóa thành công
    }
});
```

Thảo luận

